MORDETHRHEDAN — CONCEPT

> Résumé vivant de la demande. **Mis à jour à chaque échange** : nouvelle demande, choix validé, changement d'avis. Le haut du fichier décrit l'état actuel ; le journal en bas garde l'historique.

LA CHAÎNE

Streamer	Mordethrhedan
Identifiant Twitch	`mordethrhedan` — **à confirmer**
Ce qui est streamé	Jeux, dont du **speedrun** (ex. Beyond Good & Evil, Moonlighter 2)
Écran / canevas OBS	1920×1080 (supposé)
Alertes	**StreamElements** (jeton dans reglages.html) : alertes de l'overlay, objectif, ligne des derniers événements. Ses alertes Streamlabs peuvent rester s'il y tient

DIRECTION ARTISTIQUE

- **Univers** : **néon sobre**. Refaire proprement son overlay existant (cadres verts « faits sous Paint »), **sans grosses animations**.
- **Fond** : facettes sombres façon cristal avec des éclats lumineux, **recréé et généré** (à partir d'un thème Chrome trouvé en ligne), qui respirent très doucement (désactivable).
- **Couleur** : **une seule couleur d'accent, qui pilote tout** (cadres + éclats du fond). Vert par défaut, changeable selon le jeu (`couleur` dans `config.js` ou `?couleur=rouge`).
- **Cadres** : trait régulier, coins arrondis, liseré sombre, **halo discret** (léger / moyen / fort), version transparente et version « verre » fumée.
- **Police** : **Dyer** (celle de Beyond Good & Evil) **partout à l'écran** (titres, textes, chat, alertes, transitions, kit), incluse dans `css/theme.css` (aucun fichier à charger) ; accents ajoutés. Nunito seulement pour les pages d'outils (réglages, vitrine).
- **Vocabulaire des alertes** : sobre et direct (« Nouveau follow », « Nouvel abonné », « Raid ! »…). Carillon doux ; les gros événements font pulser le halo.
- **Chat** : en cartes, comme son ancien chat, avec pastilles de badges (streamer, modo, VIP, abonné).

CE QUI EST DEMANDÉ

- **Trois dispositions** (refonte d'après ses 3 croquis) :

    - **Grande** — Démarrage (compte à rebours), Pause, Fin, Cam seule : pseudo en haut, grand cadre troué (cam, image, contenu), message au centre facile à changer, **ligne des derniers événements** en bas (follow · sub · raid · série de visionnage), chat à droite.
    - **Petite** — Contenu, Speedrun : manette ou cam en haut à gauche, LiveSplit en bas à gauche (ou le chat, désactivable d'un clic), pseudo + grand cadre + derniers à droite.
    - **Jeu** — cadre au ras des bords, coins bouchés par le fond, pseudo dans un encadré posé sur le cadre (au-dessus / en dessous de la cam), **cadre de cam + pseudo en source séparée** (`sources/cam.html`) à grouper avec la cam et son compteur de morts : tout disparaît d'un raccourci OBS (pseudo compris).
- **Fond synchronisé** : `sources/fond.html` tout en bas de chaque scène, calé sur l'horloge → raccord avec le fond de l'overlay là où il n'y a pas de source.
- **Sources** : alertes, chat, objectif, **cadre néon à la taille voulue** (pour encadrer ses propres widgets), compte à rebours seul, fond seul.
- **Transitions** : balayage et volets néon (simples).
- **Chaîne Twitch** : bannière, écran hors-ligne, panneaux de bio, emotes, badges d'abonné — fait (`chaine/`), à valider.
- **Contraintes** : son widget de succès, il le gère lui-même (non pris en compte). Manette et LiveSplit : des trous encadrés.

DÉCISIONS PRISES

- 2026-09-29 — Fond généré (graine réglable) plutôt que l'image Chrome d'origine.
- 2026-09-29 — Mises en page regroupées dans `js/scenes.js`.
- 2026-09-30 — Dossier `Overlay/mordethrhedan/`, police dans `assets/polices/`, accueil et moodboard remis au format commun, flou de la barre du moodboard retiré (écran noir).
- 2026-09-30 — Refonte en 3 dispositions (Grande / Petite / Jeu). Derniers événements sans bot : sub, raid, série via les annonces du chat Twitch ; follow via la Socket API Streamlabs (clé dans `config.js › streamlabs.jeton`). Module propre à l'overlay : `js/derniers.js`.

État d'avancement

Élément	État
Moodboard	✅
Écrans et scènes	✅
Sources (dont `cam` : cadre de la cam de la scène Jeu, à grouper)	✅
Transitions (vidéos Stinger : balayage et volets, 800 ms)	✅
Kit chaîne Twitch (`chaine/` , 44 PNG dans `chaine/export/`)	✅ (à valider)
TUTO.md	✅
Refonte des scènes (3 dispositions, derniers événements, fond synchronisé)	✅ vérifiée en navigateur
Testé dans OBS / annonces Twitch et StreamElements en vrai live	⬜ (StreamElements : faux serveur seulement)

À faire / questions ouvertes

- ☐ Confirmer l'identifiant Twitch, brancher StreamElements (TUTO 6.2 : compte + jeton dans reglages.html).
- ☐ Refaire ses scènes OBS avec les nouvelles fiches (TUTO §3), dont le groupe « Cam » + raccourci.
- ☐ Valider le kit de chaîne Twitch (`chaine/kit.html`) puis l'envoyer sur Twitch (TUTO §7).

Journal

- **2026-09-29** — Captures de l'ancien overlay fournies. Précisions : widgets ignorés, fond à refaire, halo discret, couleur unique pilotant cadres et fond, police Dyer.
- **2026-09-29** — Socle (fond, cadres, config), moodboard, puis construction complète sur le modèle Espace, vidéos Stinger, tuto.
- **2026-09-30** — Harmonisation : dossier renommé, fichiers communs, accueil et moodboard au format commun.
- **2026-09-30** — Kit de chaîne Twitch créé (profil, bannière, hors-ligne, 4 panneaux, 6 emotes, 5 badges). TUTO réécrit fiche par fiche, en PDF ; `index.html` devient une vitrine.
- **2026-09-30** — Ajout de `reglages.html` (page commune à tous les overlays, aux couleurs du thème) : un formulaire qui modifie `config.js` sans toucher au reste du fichier ; libellés propres à l'overlay dans `js/reglages-champs.js` . TUTO §2.1 mis à jour. Outils communs mis à jour (variable `NAVIGATEUR` , polices locales du thème dans les PDF). PDF à régénérer sur le PC (`node outils/generer-pdf.mjs`).
- **2026-09-30** — Polices passées en local (`assets/polices/` , via `node outils/polices-locales.mjs`) : l'overlay n'a plus besoin d'internet pour s'écrire (vérifié en bloquant Google Fonts). Dyer reste à déposer à la main (Righteous la remplace).

- **2026-09-30** — Objectif réglé sur les abonnements : une pluie d'abonnements offerts compte maintenant pour tous ses cadeaux (avant : 0). Chat : mémoire des 10 dernières minutes entre scènes (`chat.memoireMinutes`), options `?chat=0` (TUTO §2). PDF lisibles hors ligne (police du thème si Nunito manque), régénérés.
- **2026-09-30** — Emotes 7TV / BTTV / FFZ : pas prévues (les chaînes n'utilisent pas ces extensions).
- **2026-09-30** — Commun : « pas de F12 dans Interagir » → journal à l'écran `?journal=1` (connexion Streamer.bot + événements bruts) ; client Streamer.bot en copie locale (plus besoin d'internet pour les alertes) ; script OBS `outils/actualiser-obs.lua` (actualiser toutes les sources, et tout seul quand config.js change) — non testé dans OBS.
- **2026-09-30** — Harmonisation avec Patagrain : panneau de bio « Matériel » ajouté (PNG exporté). Pas de bandeau dans cet overlay.
- **2026-09-30** — Demande : changer les couleurs depuis les réglages (ex. thème Halloween, orange à la place du bleu), sur tous les overlays → `config.js › couleurs` + `js/couleurs.js` (commun) ; `reglages.html` › Couleurs : nuanciers et ambiances en un clic (🎃 Halloween, 🎄 Noël, ↺ Couleurs d'origine). Vidéos de transition et kit à refaire après un changement de couleurs.
- **2026-09-30** — TUTO mis à jour : section « Les scripts » (installer Node.js et ffmpeg, lancer un script, tableau de tous les scripts dont actualiser-obs.lua), section 6 complète (chat, Streamer.bot pas à pas, journal ?journal=1, événements, dons, objectif, tests), réglages via reglages.html partout, options ?journal=1 / ?chat=0.
- **2026-09-30** — Demande : une option `?cam=0` (pas de webcam) → ajoutée (plus de cadre de cam). Puis « cam et tout, on devrait pouvoir le régler depuis les réglages » → `config.js › options` + `js/options.js` (commun) ; `reglages.html` › **Options des scènes** : le coin de la webcam de la scène Jeu (ou « Pas de webcam ») et le chat de chaque écran. Une option écrite dans l'adresse d'une source OBS passe avant.
- **2026-09-30** — Bug signalé : en Cam seule, remettre le chat (et les couleurs) dans les réglages ne marchait pas après actualisation → `js/options.js` ajoutait `?chat=0` à l'adresse, et l'actualisation d'OBS rechargeait cette adresse modifiée. Corrigé : les options ajoutées par les réglages sont notées (`depuisReglages=…`) et retirées au chargement suivant. Couleurs : l'écriture et l'affichage vérifiés (Halloween puis couleurs d'origine) ; pas testé dans OBS.
- **2026-09-30** — Réglages : **« Mes ambiances »** (section Couleurs) — nommer les couleurs affichées puis 💾, gardées dans `config.js › ambiances` (ex. Batman) ; un clic les remet, × les supprime. TUTO §9 mis à jour. Aperçu des alertes dans reglages.html au vrai style de l'overlay (`apercuAlerte`, comme Patagrain).
- **2026-09-30** — **Refonte des scènes demandée (3 croquis)** : ① « grande scène » (démarrage/pause/fin/cam/contenu) : trou dans le grand cadre, message facile à changer, pseudo en haut, cadre central un peu plus bas, ligne en bas « dernier follow / sub / raid / série de visionnage » ; ② « petite scène » (contenu/speedrun) : trous grand cadre (jeu/fenêtre), haut gauche (manette/cam), bas gauche (LiveSplit, ou chat désactivable) + pseudo et derniers ; ③ Jeu : cadre au plus près des bords, coins bouchés par le fond, cam dans les 4 coins, pseudo dans un petit encadré sur le cadre (au-dessus si cam en haut, en dessous si cam en bas), **cadre de cam = source séparée** (à grouper avec la cam pour la masquer d'un raccourci). Partout : **fond synchronisé** visible quand il n'y a pas de source. **Alertes : Streamlabs** (pas de bot) ; widget de succès géré par lui.

- **2026-09-30** — Refonte faite : `js/scenes.js` (3 dispositions), `js/derniers.js` (nouveau), `sources/cam.html` (nouveau), `chat.html?place=petite|grande`, fond calé sur l'horloge (`js/fond.js`), réglages (Derniers événements, chat du Speedrun), TUTO §3 réécrit + 4.7 + 6.1 bis, vitrine et moodboard. Vérifié : toutes les pages sans erreur, normal et `?test=1`. Pas testé : OBS, vraies annonces Twitch / Streamlabs.
- **2026-09-30** — Choix : le pseudo de la scène Jeu **disparaît avec la cam** → déplacé de `scenes/jeu.html` vers `sources/cam.html` (`?pseudo=0` pour l'enlever) ; sans webcam (`?cam=0`), l'overlay Jeu l'affiche en haut à gauche. Vérifié en navigateur (cam en haut, en bas, sans cam).
- **2026-09-30** — Standardisé dans les 4 overlays : **heure fixe** du compte à rebours (`demarrage.heure` / `?heure=20:30`, réglages › Démarrage › « … ou heure fixe ») et **étiquettes de placement** (`afficherZones` / `?zones=1`, réglages › Options des scènes). TUTO §2 et §3.1 mis à jour.
- **2026-09-30** — Pseudo du cadre de cam réglable aussi dans `reglages.html` › Options des scènes (`config.js › options.cam.pseudo`).
- **2026-09-30** — Demande : des sons différents selon l'alerte, pour les reconnaître en jouant → `js/son.js` : un son par alerte (follow, abonnement, réabonnement, abonnement offert, pluie d'abonnements, bits, raid, don, objectif), dans le thème (tableau dans TUTO 6.8). Et ses propres fichiers possibles : dossier `sons/` + `config.js › alertes.sons` ; `reglages.html` › **Sons des alertes** avec ▶ pour écouter (« aucun » = silence). Vérifié dans le navigateur (sons rendus hors ligne, fichier perso joué) ; pas écouté dans OBS.
- **2026-10-01** — Bug signalé : l'encadré du pseudo au-dessus de la cam était tronqué (collé au bord de l'écran, contour et halo coupés, touchant le cadre de cam). Corrigé : encadré à 4 px du bord, un peu plus fin (texte 26 px), désormais entier avec un espace avant la cam ; position de la cam inchangée. Vérifié en capture (cam en haut et en bas).
- **2026-10-01** — Demande : « quand je donne un nouveau dossier, son config est écrasé » → réglages perso séparés : `config.js` = valeurs par défaut (remplacé à chaque mise à jour), `mes-reglages.js` = ce que le streamer change dans reglages.html (jamais livré, gardé lors d'une mise à jour, `.gitignore`). Migration unique : reglages.html › « 📩 Reprendre les réglages d'un ancien config.js ». TUTO §2.1 (« Quand tu reçois une nouvelle version »). Pas testé dans OBS.
- **2026-10-01** — Demandes : « la cam de l'overlay bouge mais pas la source caméra dans OBS », « un script qui place toutes les cams sur toutes les scènes », « Options des scènes trop fouillis », « la cam de Jeu à un X/Y précis, en gardant les préréglages, et voir leurs coordonnées ». → `js/zones.js` (les zones de la webcam, seule source des chiffres) + `Options.cam()` (commun) ; scène Jeu : préréglage, position perso `{ x, y, l, h }` (`?cam=x,y,l,h`) ou pas de webcam. `reglages.html` › Options des scènes refait : un tableau scènes × éléments avec interrupteurs, et un bloc « La webcam de la scène Jeu » (préréglages avec coordonnées, position perso, plan où l'on fait glisser la cam, zones fixes des autres scènes). Script OBS `actualiser-obs.lua` : bouton « Placer les webcams sur toutes les scènes (et les ajouter là où elles manquent) » + replacement automatique quand les réglages changent. Testé hors OBS (faux OBS, 1440p, groupes, carte d'acquisition non touchée) ; **pas testé dans le vrai OBS**.
- **2026-10-02** — Bug signalé : en scène Jeu, le pseudo au-dessus de la cam était encore tronqué dans OBS (le grand cadre de la scène, s'il passe par-dessus le groupe de la cam, en recouvrait le haut). Corrigé à sa demande en **baissant la cam par défaut** : préréglages Jeu à 72 px du haut / du bas (`js/zones.js` : y 72 et 708, x inchangés), encadré du pseudo posé entièrement à l'intérieur du grand cadre (18 px du bord). Vérifié en capture dans le pire ordre des sources (scène Jeu par-dessus la cam),

cam en haut et en bas. TUTO 3.6 mis à jour. Pas testé dans OBS. Note : ffmpeg est désormais bloqué par une stratégie Windows (contrôle d'application) sur ce PC → `generer-transitions.mjs` ne pourra plus encoder tant que ce n'est pas levé.

- **2026-10-02** — Demande : « ne plus envoyer de zip à la main ; un serveur sur mon VPS ; un push = mis à jour ; et quand eux modifient ? » → dossier `serveur/` (à la racine du dépôt) : le VPS publie une page de téléchargement + un zip et un `version.json` par overlay, refaits à chaque push (webhook GitHub ou GitLab, + vérification horaire), installés par `serveur/installer.sh` (notice : `serveur/LISEZMOI.md`). Côté overlay : bouton **Mettre à jour l'overlay** dans le script OBS `actualiser-obs.lua` (ou `mettre-a-jour.cmd`) : dernière version installée, `mes-reglages.js` gardé, ancienne version dans `sauvegardes\`. `reglages.html` › **Mises à jour** (adresse du serveur). TUTO 2.1, 2.8, scripts, dépannage. Testé : installation en conteneur, webhook, mise à jour avec PowerShell 7 ; **pas testé** : sur le vrai VPS, sous Windows, dans OBS.
- **2026-10-02** — Choix : le dépôt publié par le serveur sera sur GitLab (https://gitlab.com/Bastien.Lambour/overlays, branche main), webhook GitLab. `serveur/installer.sh` et `serveur/LISEZMOI.md` réglés dessus par défaut.
- **2026-10-02** — Le VPS héberge déjà d'autres sites (Nginx, certificat bastien-lambour.fr) → le serveur des overlays sera sur **https://overlays.bastien-lambour.fr** (même certificat, 80 → 443), dans `/var/www/overlays` (`depot/` privé, `site/` servi) ; `installer.sh` n'ajoute qu'un fichier Nginx et ne touche à aucun autre site. `miseAJour.adresse` = cette adresse. À faire : l'enregistrement DNS `overlays` → 217.154.115.223.
- **2026-10-02** — Demande : « une barre d'objectif pour les follows aussi ? avec les vrais nombres de Twitch, pas à la main ; plutôt un interrupteur : la barre affiche les followers OU les abonnés, chacun son compteur et son objectif » → **une seule barre**, `config.js › objectif` : `affiche` ("follow" ou "sub"), `automatique`, et `follow` / `sub` (titre, cible, départ chacun) ; les deux compteurs tournent toujours (bascule = barre tout de suite juste) ; `?objectif=sub|follow` sur une source. Les vrais totaux viennent de **Streamer.bot** : action « Overlay – Compteurs » (`outils/streamerbot-compteurs.cs`, à coller une fois, TUTO 6.6), lancée par l'overlay à la connexion et toutes les 5 minutes. Anciens réglages (`objectif.type/cible/depart`) repris tout seuls. Vérifié dans le navigateur (follow/sub, mode test, ancien mes-reglages.js) ; **pas testé** : le code C# dans un vrai Streamer.bot (pas de Streamer.bot ici).
- **2026-10-02** — Demande : « la police Dyer (TTF fourni) PARTOUT : bannière, hors-ligne, transitions, cadres des follows/raids… et incluse pour ne pas attendre 10 s à l'actualisation » → Dyer pour tout ce qui passe à l'écran (`--f-ecran`, posé sur `#ecran` et le kit `.mn`), sans faux gras (`font-synthesis: none`, elle n'a qu'une graisse) ; **incluse dans** `css/theme.css` (data: woff2, 15 Ko, `font-display: block`) : pas de fichier à part ni de passage par une autre police. Le TTF n'avait **aucune lettre accentuée** : ajoutées (é è ê ë à â ä î ï ô ö ù û ü ç, majuscules et minuscules, + espaces insécables) en assemblant les lettres et accents de la police (`assets/polices/Dyer.woff2`). Réglages et vitrine restent en Nunito (lisibles). Vidéos de transition et PNG du kit refaits. Vérifié dans le navigateur (toutes les pages, police chargée) ; **pas vu dans OBS**.
- **2026-10-02** — Demande : « au lieu de Streamer.bot, StreamElements ? » puis « oui, on remplace Streamer.bot ! » → **Streamer.bot retiré**, tout passe par **StreamElements**, par internet (rien à installer) : `js/evenements.js` (commun) se branche à `wss://astro.streamelements.com` avec le jeton du compte (« JWT Token », `reglages.html` › StreamElements, gardé dans mes-reglages.js), écoute `channel.activities` (follow, sub, réabonnement, cadeau, pluie de cadeaux annoncée une seule fois,

bits, raid, dons) et `channel.session.update` (vrais totaux followers / abonnés pour l'objectif, lus aussi au branchement), se rebranche tout seul. Plus d'action C# ni de client Streamer.bot ; les anciens réglages streamerbot sont oubliés au prochain Enregistrer. TUTO §6 réécrit (6.2 brancher StreamElements, 6.3 journal, 6.5 dons par la page de dons StreamElements, 6.6 vrais nombres automatiques). Testé dans le navigateur avec un **faux serveur StreamElements** (chaque type d'événement, compteurs, objectif atteint, reconnexion) ; **pas testé** avec le vrai StreamElements ni dans OBS.

- **2026-10-02** — Question : « pourquoi une clé Streamlabs ? » (elle servait au seul dernier follow de la ligne du bas) → **option A choisie** : tout sur StreamElements comme les autres. `js/derniers.js` prend follow, sub et raid de StreamElements (via `js/evenements.js`, chargé maintenant sur toutes les scènes et `sources/cam.html`), la série de visionnage reste lue dans le chat ; clé Streamlabs retirée (config, réglages ; l'ancienne est oubliée au prochain Enregistrer). Ses alertes Streamlabs peuvent rester s'il y tient. Testé avec un faux serveur StreamElements.
- **2026-10-02** — Demande : « un bouton pour regénérer les transitions et tout, depuis les réglages ? » → impossible depuis une page web (un navigateur ne lance pas de programme) : choix **depuis OBS**. Script OBS (`outils/actualiser-obs.lua`) : boutons **Refaire les vidéos de transition** et **Refaire les images du kit Twitch** (`outils/refaire.ps1` , en arrière-plan ; installe Node.js et ffmpeg au 2e clic s'ils manquent ; les transitions Stinger de l'overlay sont décrochées pendant la fabrication puis rechargées ; le dossier du kit s'ouvre à la fin). `reglages.html` le rappelle après un changement de couleurs, du nom ou des textes du kit. TUTO 2.8 (+ §5, §7, §9). Testé avec PowerShell sous Linux (faux cmd / winget / explorer : vraies vidéos et images refaites) et un faux OBS ; **pas testé sous Windows ni dans un vrai OBS**.
- **2026-10-06** — Demande : « un fond et une bannière pour la page de dons StreamElements (640×200 et 1920×1080), régénérables comme le kit, et un panneau de bio Commandes » → kit : nouveau groupe **Page de dons StreamElements** (`dons-banniere.png` , `dons-fond.png` : deux cadres de verre, monogramme et cœur néon ; le milieu du fond reste libre pour le formulaire), textes dans `config.js › chaine.dons` (titre, texte ; reglages.html › Kit), panneau **Commandes** ajouté à `chaine.panneaux` . TUTO §7 : pas à pas StreamElements (Revenue › Tipping settings) et description du panneau Commandes.
- **2026-10-06** — Demande : « un panneau Discord pour la bio Twitch, et les réseaux classiques » → panneaux **Discord, YouTube, TikTok, Instagram, X (Twitter)** ajoutés à `chaine.panneaux` ; chaque nom de réseau prend son logo (fichier commun `chaine/reseaux.js`, logos simplifiés d'une couleur, aux couleurs du thème ; aussi Twitch, Kick, Bluesky). TUTO §7 : lien de l'image vers le réseau.
- **2026-10-07** — Demande : « convertir cadre.html en image ? » → `outils/cadre-image.mjs <l>x<h> [verre]` : PNG transparent du cadre néon (halo compris, 50 px de marge) dans `assets/cadres/` , aux couleurs du moment ; TUTO §4.4 et tableau des scripts. Outil commun `capturer.mjs` : options `--transparent` , `--decoupe=x,y,l,h` , `--fichier=` .
- **2026-10-12** — Demandes : « générer le panneau Planning depuis les réglages » et « un panneau Mes créations / Mes sites » → le panneau **Planning** devient le planning de la semaine (jours, heures, jeu en option : « Mercredi | 20h30 | Minecraft ») dès que `reglages.html › Kit › Planning` est rempli (vide = panneau titre comme avant) ; panneau **Mes créations** ajouté (pinceau ; « Mes sites » = globe, via `chaine/reseaux.js`). Titres longs des panneaux ajustés.

- **2026-10-12** — Question : « dans les réglages, tout est propre et organisé ? » → section **Kit de chaîne Twitch** rangée (une aide en tête qui renvoie au bouton OBS, textes, page de dons, panneaux, planning) ; ancien champ **Réseaux** retiré (plus utilisé : les logos viennent de la liste des panneaux ; `reglages.js` l'oublie dans les anciens réglages).
- **2026-10-12** — Demande : « un bouton dans les réglages pour régénérer le planning, choisir la semaine (« planning semaine 8 ») sans effacer celui qu'on a » → sous le champ Planning de `reglages.html` : calendrier de semaine + 🖼️ **Fabriquer l'image du planning** (la page enregistre, dépose `outils/demande.txt` ; le script OBS lance `refaire.ps1 -Quoi planning -Semaine …` et répond dans `outils/demande-resultat.txt` ; la page montre l'image). Une image par semaine dans `chaine/export/plannings/` (sous-titre « Semaine 8 · du … au … »), les anciennes restent.
- **2026-10-12** — Retour : « le bouton du planning ne fonctionne pas » + « le fabriquer SANS OBS ouvert » → correctif (la page ne redemandait la permission du dossier qu'après l'enregistrement, ce que le navigateur peut refuser : elle vérifie d'abord si elle l'a déjà) et message clair quand personne ne prend la demande ; nouveau `fabriquer-images.cmd` (double-clic, `outils/fabriquer.ps1`) : traite la demande de la page sans OBS, ou menu planning d'une semaine / images du kit / vidéos de transition.
- **2026-10-13** — Demande (via la session CodexVTT, validée par l'utilisateur) : une **scène JDR** pour Mordeth, qui JOUE dans les parties de Patagrain (MJ) sur CodexVTT. Moteur commun `js/partie.js` (+ `codex.js`, `socket.io.min.js` dans `_modele`) : table construite depuis CodexVTT, réglage « mon personnage » (sa carte en avant, ses jets en grand). **Moodboard** `design/moodboard-jdr.html` : 3 dispositions néon (A table en bas, B colonne de vie + fil des jets, C joueur en vedette avec sa webcam), maquettes animées `design/jdr-maquette.html?d=A|B|C`. En attente du choix.
- **2026-10-13** — Choix : **disposition A** (« table en bas ») → `scenes/jdr.html` construite (`css/jdr.css`, `js/jdr.js` : cartes néon, compteur néon des jets, « il parle »), branchée sur le moteur commun (`js/partie.js`, `codex.js`, `voix.js`) : sa carte « toi » et ses jets en grand, les autres en petit, carte du MJ (Patagrain), la carte de celui qui parle sur Discord s'anime (même bot « voix » que Patagrain : réglages › Scène JDR › « Qui joue qui sur Discord »). TUTO §3.8. Demande suivante : **un personnage = une fiche d'apparence commune**, dessinée par chaque overlay à son style (Xaieysi créé chez Patagrain → le même chez Mordeth, en néon), catalogue commun dans le dépôt, un créateur de personnage ; d'abord un moodboard de style de perso pour Mordeth, John Von Gurt et Ambries.
- **2026-10-14** — Fiche d'apparence commune des personnages : `js/apparence.js` (les traits → les formes d'un buste, expressions, formes comme Trayss dragon) et catalogue commun `js/persos.js` (Xaieysi en exemple + Namerya, Zokhran, Trayss). **Moodboard** `design/moodboard-persos.html` : deux directions de style de personnage pour la chaîne (`js/persos-style.js`). En attente du choix.
- **2026-10-14** — Choix du style des personnages : **« Trait néon »** (`js/persos-style.js`). Les persos du catalogue commun apparaissent dessinés sur les cartes JDR (bouche qui bouge quand ils parlent, K.O. à 0 PV, forme de CodexVTT). Demande : garder partout les **scénettes au-dessus du chat** → `js/scenette.js` (le perso joue son geste pendant le jet, puis réagit au résultat). **Créateur de personnage** `createur.html` (fiche à copier/télécharger pour le catalogue commun). TUTO §3 JDR complété.
- **2026-10-15** — Harmonisation des 4 overlays : **page de réglages** refaite pareil partout (un réglage par ligne, sommaire des sections en haut, réglages avancés repliés, page plus large ; un tableau trop large devient des fiches : plus de défilement de côté). Scène JDR : **seuls les personnages et joueurs choisis**

ont une carte, mais **tous les jets sont joués** (un personnage sans carte apparaît avec son apparence ou son portrait). Oreille droite des personnages réparée. Lien « Créateur de personnage » dans la vitrine.

- **2026-10-15** — Scène JDR branchée sur la **boîte à outils de CodexVTT** (`js/codex-stream.js`) : l'overlay ne dessine plus lui-même les personnages, il demande le dessin à la boîte à outils (apparence enregistrée dans CodexVTT, sinon le catalogue). Nouveau réglage **Style des personnages** (reglages.html › Scène JDR) : Néon, Sticker, Écusson, Vitrail, Conte ou Aplat. Style par défaut : Néon (le même qu'avant).
- **2026-10-15** — Nouveau réglage **Personnages en pied dans les scénettes** (décoché par défaut) : pendant un jet, le personnage apparaît en entier et joue le geste avec ses bras et ses jambes.
- **2026-10-15** — Personnages en pied : ils sont maintenant sur le **squelette du bouffon de Patagrain**, avec ses gestes et la mise en scène de son arène (le même moteur pour toute la famille et pour les widgets de CodexVTT). Option « Personnages en pied dans les scénettes » inchangée (décochée par défaut).
- **2026-10-15** — Le créateur de personnage est retiré de l'overlay (et de la vitrine) : il est dans CodexVTT › labo › Personnages. Traits du visage plus fins.
- **2026-10-20** — Les personnages de la scène JDR sont maintenant dessinés et animés par **CodexVTT lui-même** (`js/codex-stream.js` = le lecteur de CodexVTT) : les retouches de dessin se font dans CodexVTT et arrivent ici avec les mises à jour. Pendant un jet, le personnage bouge bras et jambes (avant : son image bougeait d'un bloc).
- **2026-10-21** — Scène JDR : les **jets de dés ne sont plus animés dans l'overlay** (plus de compteur ni de personnage au-dessus du chat) : c'est une **source OBS de CodexVTT** (« Animations des jets »), posée en grand sur la table, que les spectateurs voient et pas les joueurs (TUTO, fiche JDR). Les cartes montrent le personnage **en mi-corps à gauche**, sans cercle. L'ancien **bot Discord** est retiré (réglages et tuto) ; « qui parle » reviendra plus tard par Discord StreamKit. Réglages « Qui joue qui sur Discord » → « Les personnages de la partie ».
- **2026-10-21** — Scène JDR **harmonisée** dans les 4 overlays : la scène n'a plus que le décor, le titre de la campagne et le chat. La table (capture), les **cartes des personnages** et les **animations des jets** viennent de **CodexVTT**, en sources OBS à part, au thème de la chaîne ; les personnages, le MJ et les sous-titres se choisissent dans CodexVTT (Sources pour OBS). Réglages JDR : seulement le nom de la campagne. Tuto (fiche JDR) réécrit pas à pas. Police Dyer pour les sources de CodexVTT : `obs/police-dyer.css`. Moodboard et maquette JDR retirés (obsolètes).
- **2026-10-21** — **Qui parle** (Discord StreamKit, sans bot, facultatif) : quand un joueur parle sur Discord, le personnage de sa carte parle aussi. Une source Discord posée sur les cartes, avec un CSS fabriqué par CodexVTT d'après les identifiants Discord des joueurs. Pas à pas dans le tuto (fiche JDR). Pas encore essayé sur le vrai Discord.
- **2026-10-21** — Demande : « il n'y a pas de tuto JDR comme chez Patagrain » → `TUTO-JDR.md` / `.pdf` (le même plan que celui de Patagrain : CodexVTT, les sources dans OBS, qui parle, la partie, le dépannage), lien dans la vitrine ; la fiche JDR du tuto général le résume.
- **2026-10-21** — Scène JDR : **la même disposition dans les 4 overlays** (table en 60, 60 ; cartes en 60, 896 ; titre et chat à droite en x 1540) : les mêmes chiffres dans tous les tutos, et une scène OBS faite pour un overlay marche pour un autre.

- **2026-11-02** — CodexVTT ne sert plus les overlays (décision de l'utilisateur). Scène JDR refaite sans lui : la table de jeu capturée (CodexVTT ou un autre site), et **les cadres des personnages dessinés par l'overlay** (image, nom, sous-titre, PV, une ressource au nom réglable, « à terre », « son tour »), un cadre par place cochée dans `reglages.html`. Les PV se changent pendant la partie dans le **pupitre** (`pupitre-jdr.html`, un dock d'OBS). Tutos JDR réécrits. Pas encore essayé dans OBS.
- **2026-11-03** — Les images des personnages se posent **dans OBS** : la case du portrait de chaque cadre est vide et transparente, on y met une source Image (position et taille dans le tuto JDR, section 3.4, et sur `scenes/jdr.html?apercu=1`). La colonne « Image » des réglages est retirée.