

# Spécifications de l'Application de Tirage au Sort Multiplateforme

## Introduction et Contexte

Organiser des concours sur les réseaux sociaux (Instagram, Facebook, Twitter/X, TikTok, etc.) est un moyen populaire d'engager une communauté, mais garantir un tirage au sort (TAS) *fiable, équitable et transparent* représente un défi. De nombreuses marques et community managers réalisent manuellement ces tirages, vérifiant laborieusement que chaque participant respecte les règles (abonnement aux comptes, mentions d'amis, partage en story...). Cette application vise à **automatiser** le processus de tirage au sort de concours sur plusieurs plateformes, tout en assurant la **conformité légale (RGPD)**, la **transparence vis-à-vis des participants** et une **expérience utilisateur (UX)** fluide.

### Objectifs principaux :

- Proposer un outil *multiplateforme* (web et mobile) pour tirer au sort des gagnants de concours de manière aléatoire et équitable.
- Faciliter la vérification des conditions de participation (abonnements, mentions, partages en story, etc.) afin d'éviter les abus et tricheries.
- Offrir une expérience simple, rapide et moderne, avec un système de **crédits clair** (pas de « coins » opaques) et des options d'abonnement pour un usage régulier.
- Garantir la confiance des utilisateurs grâce à des fonctionnalités de preuve du tirage (enregistrement vidéo de l'animation, certificat de résultat partageable, etc.) <sup>1</sup> <sup>2</sup> .

Ce document décrit en détail les fonctionnalités, les choix techniques, la tarification envisagée et d'autres aspects (design, conformité, stratégie de lancement) pour le développement du service de tirage au sort.

## Plateformes Ciblées et Technologies

### Disponibilité Web et Mobile

Dès le lancement, l'application sera disponible :

- **Sur le web (navigateur)** : une application web responsive, accessible sans installation, pour ne pas limiter les usages et permettre aux utilisateurs sur ordinateur ou mobile d'y accéder facilement.
- **Sur mobile (Android et iOS)** : applications mobiles natives développées en **React Native** afin d'assurer une base de code commune et une expérience cohérente sur Android et iOS. Ce choix garantit un développement plus rapide *multiplateforme* tout en offrant des performances proches du natif.

**Avantage multiplateforme** : un community manager pourra aussi bien utiliser le service depuis son PC au bureau que depuis son smartphone lors d'un événement, sans discontinuité.

## Stack Technique Préférentielle

- **Backend (serveur)** : une API RESTful complète construite en **Node.js** (JavaScript/TypeScript). Node.js est choisi pour sa performance en temps réel (utile pour récupérer de gros volumes de commentaires rapidement) et l'écosystème riche (paquets pour interagir avec les API des réseaux sociaux, etc.). Le backend exposera des endpoints sécurisés pour toutes les fonctionnalités (tirer un gagnant, vérifier des critères, gérer les comptes utilisateurs...).
- **Base de données** : une base de données relationnelle (par ex. PostgreSQL) ou NoSQL selon les besoins, hébergée en Europe (voir section RGPD). Elle stockera les comptes utilisateurs, crédits restants, historiques de tirages, etc.
- **Frontend Web** : développé en **React.js** (framework front-end moderne) pour une interface utilisateur réactive et dynamique, suivant les standards UX modernes. La partie web et la partie mobile partageront certains composants/contenus (via une API commune).
- **Mobile** : via **React Native**, permettant de réutiliser la logique métier du frontend. L'interface sera adaptée aux conventions iOS et Android tout en conservant un style unifié.
- **Hébergement** : une infrastructure cloud scalable (type AWS, Azure ou autre), avec des serveurs en Europe pour la conformité. On prévoira une architecture évolutive (par ex. microservices modulaires) si la charge augmente, mais initialement un backend monolithique Node suffira. Un CDN pourrait être utilisé pour distribuer la web app et le contenu statique rapidement.

(À confirmer ultérieurement :) tests automatisés (CI/CD) pour fiabiliser les déploiements, surveillance des performances lors des tirages comportant des milliers de participants, etc., feront partie des considérations techniques.

## Inscription, Authentification et Comptes Utilisateurs

### Utilisation avec ou sans Compte

L'application offrira deux modes d'utilisation pour s'adapter au plus grand nombre :

- **Utilisation ponctuelle sans compte** : Un utilisateur qui souhaite effectuer un tirage unique pourra le faire sans créer de compte. Il achètera alors un **crédit unique** "one-shot" pour réaliser son tirage au sort. *Avantage* : simplicité pour l'utilisateur occasionnel (pas de friction de création de compte).
- **Utilisation avec compte (requis pour usages répétés)** : Dès qu'un utilisateur souhaite acheter un pack de crédits (plusieurs tirages) ou souscrire un abonnement, la création d'un compte sera **obligatoire**. En effet, il faudra associer ces crédits à son profil pour qu'il puisse les réutiliser plus tard.

### Processus de Création de Compte

La création d'un compte utilisateur nécessitera :

- **Adresse e-mail et mot de passe**, ou éventuellement l'option d'un **login via Google/Facebook** (à étudier, mais l'email reste nécessaire pour la communication).
- **Vérification de l'e-mail** : une validation par lien envoyé par courriel sera requise pour activer le compte, garantissant l'authenticité de l'adresse fournie et permettant d'envoyer les reçus d'achat, preuves de tirage, etc.

Chaque compte correspond en principe à **une entité (marque/entreprise)** ou un particulier gérant ses propres concours. Nous limiterons dans les CGU une utilisation commerciale détournée (voir “Offres entreprise” plus bas) pour éviter qu’un seul compte serve à de multiples sociétés non liées.

## Gestion des Comptes et Limitations

- **Récupération de mot de passe** : via email en cas d’oubli.
- **Informations stockées** : e-mail, hash du mot de passe, éventuellement nom/prénom ou nom de la société, et préférences (langue, etc.). **Pas de données sensibles inutiles** stockées pour minimiser l’impact RGPD.
- **Sécurité** : mots de passe stockés chiffrés (hash + salt), communications chiffrées (HTTPS). Option d’authentification à deux facteurs (2FA) à considérer si des comptes gèrent des crédits importants.
- **Utilisateurs par compte** : Par défaut, un compte = un utilisateur (un login). Pour les besoins d’équipe (plusieurs personnes d’une même entreprise voulant accéder au compte), on pourra ajouter plus tard la gestion d’utilisateurs secondaires ou un partage sécurisé de l’accès. Ce n’est pas prioritaire pour le MVP.
- **Profil de l’utilisateur** : pourra indiquer la/les plateformes principales utilisées (Instagram, Facebook...) et lier ses comptes de réseaux sociaux (via OAuth) pour faciliter les tirages (voir section intégration réseaux sociaux).

## Système de Crédits et Tarification

### Principe des Crédits

La tarification sera **claire et transparente** : **1 crédit = 1 tirage** (pour un post donné, avec X gagnants tirés). Cette conversion simple évite toute confusion – **pas de système de “pièces” virtuelles** où l’utilisateur doit convertir entre des coins et des tirages. Ce choix vise la simplicité UX : l’utilisateur sait qu’en achetant 5 crédits, il pourra organiser 5 tirages au sort, point barre. *(De nombreux concurrents utilisent aussi ce principe de crédit unitaire plutôt que des monnaies virtuelles compliquées)* <sup>3</sup> .

**Important** : Un tirage correspond généralement à la sélection d’un ou plusieurs gagnants pour **un seul post** (ou un ensemble multi-posts dans le cas de concours multi-plateformes, cf. section dédiée). Si un utilisateur souhaite relancer un tirage sur le même post (par exemple refaire le tirage parce que le gagnant initial ne remplit pas une condition ou ne répond pas), cela **consommara à nouveau un crédit**. L’application pourra toutefois offrir la possibilité de tirer des *gagnants suppléants* dans le même tirage initial pour pallier ce cas sans dépenser plus (ex : tirer 1 gagnant + 2 suppléants en un tirage).

### Achats de Crédits : Options Disponibles

Nous proposerons plusieurs façons d’acheter des crédits pour convenir à tous les profils :

- **Crédit à l’unité (one-shot)** : idéal pour quelqu’un qui a un besoin ponctuel. Par exemple, **1 crédit pour 1,99€** (tarif indicatif). Aucune création de compte requise pour cet achat unitaire, l’utilisateur paie et effectue immédiatement son tirage. *Cependant*, s’il souhaite en refaire un plus tard, il ne pourra pas “garder” ce crédit inutilisé puisque sans compte il faut consommer de suite. Cette offre sert surtout d’appel (et évite de rebuter les utilisateurs avec une inscription obligatoire).
- **Packs de crédits (multi-tirages)** : pour les utilisateurs réguliers qui ne veulent pas d’abonnement mais plusieurs tirages à utiliser quand bon leur semble. **Création de compte requise** ici, afin d’attacher les

crédits achetés au compte. On peut s'inspirer des pratiques concurrentes pour déterminer les packs : par exemple, *Starter Pack* 5 crédits, *Pro Pack* 10 crédits, *Max Pack* 20 crédits, avec des tarifs dégressifs par tirage. Par exemple : 5 crédits pour 8€ (soit 1,6€ chacun), 10 pour 15€ (1,5€ chacun), 20 pour 28€ (1,4€ chacun), etc. *(Ces chiffres seront à valider en fonction des coûts et de l'étude de marché.)* À titre de comparaison, une application similaire sur iOS propose des packs de tirages à 2,99€, 5,99€, 7,99€, 9,99€ etc <sup>4</sup>. Notre stratégie est de rester compétitif tout en assurant la rentabilité.

- **Abonnement illimité** : voir section suivante sur abonnements, destiné aux power-users (agences, grandes marques).

**Pas de coûts cachés** : L'application sera téléchargeable gratuitement. L'utilisateur saura dès le départ qu'un tirage nécessite un crédit payant. On veillera à être transparent pour éviter les frustrations du type « *Gratuit* » *affiché mais paiement obligatoire découvert après coup* (critique faite à certaines applis <sup>5</sup>). Chaque option de paiement affichera clairement ce qui est inclus.

## Méthodes de Paiement et Plateformes

- **Sur Android et Web** : intégration de **Stripe** pour les paiements par carte bancaire, Google Pay, etc. Stripe offre une solution sécurisée, rapide à intégrer, compatible web et mobile Android, avec une gestion des paiements uniques et abonnements.
- **Sur iOS (App Store)** : obligation d'utiliser les **In-App Purchases (IAP)** d'Apple pour tout achat de contenu numérique dans l'application. Apple prenant une commission (~30% pour la plupart des dev, possiblement 15% si éligible au programme Small Business), **les prix dans l'app iOS seront ajustés d'environ +30%** pour compenser ces frais Apple. Par exemple, si un pack de 5 crédits vaut 8€ sur web/Android, il pourrait être listé ~10,49€ via Apple. Cette différence sera communiquée de façon subtile aux utilisateurs iOS. *En effet, nous devons respecter les guidelines Apple qui interdisent de pousser l'utilisateur à acheter en dehors de l'application, mais rien n'empêche d'indiquer que "Nos packs sont également disponibles sur notre site web".* L'idéal est d'inciter les utilisateurs à passer par le site ou Android pour bénéficier du meilleur tarif, sans enfreindre les règles d'Apple (qui sanctionne toute mention explicite d'un prix moins cher ailleurs).

**Exemple de communication permise** : sur l'écran d'achat iOS, afficher "Astuce : vous pouvez aussi acheter des crédits via notre site web." (Apple tolère généralement l'indication d'une alternative sans mentionner les prix). L'utilisateur avisera. Sur nos pages web publiques ou documentation, on peut clairement dire que les prix *hors iOS* sont inférieurs puisque sans la commission Apple.

## Politiques de Tarification et Transparence

Nous adoptons une tarification *forfaitaire par tirage* plutôt que de complexes calculs en fonction du nombre de participants ou de critères. **Pas de frais variables selon 500, 1000 ou 5000 participants, ni selon 1 ou 5 gagnants** : cela serait perçu comme peu clair ou comme des "frais cachés". Au contraire, l'utilisateur sait qu'un crédit couvre un tirage complet, quelles que soient les conditions (dans la limite technique de la plateforme, voir plus loin). Cette simplicité nous différencie de certains concurrents qui annoncent "à partir de 1,99€" puis montent le prix en fonction des options, ce qui peut frustrer les clients.

**Cas des concours de très grande ampleur** : Nous définirons tout de même des limites techniques raisonnables pour un tirage standard – par ex. "un crédit couvre jusqu'à 50 000 commentaires" – simplement car extraire et traiter des millions de commentaires a un coût significatif. La plupart des

concours clients resteront en deçà (rarement plus de quelques dizaines de milliers de participants). Si un client prévoit un tirage sur un volume énorme (plusieurs centaines de milliers de commentaires), nous pourrions prévoir un **forfait spécial ou une consommation de crédits multiples**. Mais cela sera l'exception et géré de manière transparente (par ex. affichage d'un message "Ce concours dépasse X participants, veuillez utiliser Y crédits ou contacter notre support entreprise"). En règle générale, nous voulons **éviter de facturer au volume** pour ne pas pénaliser le succès des concours de nos clients, tout en protégeant nos coûts serveurs.

## Offre d'Essai Gratuite

Pour encourager l'adoption, on peut offrir une **utilisation gratuite limitée** : par exemple, *1 tirage gratuit* pour chaque nouveau compte (limité à un post avec  $\leq 200$  entrées). Cela permet de tester le service sans friction. Ce modèle est courant : Simpliers offre un tirage gratuit jusqu'à 200 entrées par post <sup>6</sup>, Osortoo permet un tirage gratuit jusqu'à 50 commentaires <sup>7</sup>. Nous pourrions aligner notre offre d'essai sur ~100-200 commentaires max. **Objectif** : prouver la valeur de l'outil, puis convertir en achat de crédits pour des concours plus grands. Un compteur dans l'UI indiquera si le post dépasse la limite gratuite, invitant alors à acheter un crédit.

() Limite technique due au fait qu'Instagram restreint l'accès public à tous les commentaires. Sans authentification, souvent seuls les derniers ~150 commentaires sont accessibles <sup>8</sup>. Avec un compte lié et l'API officielle, on peut tout récupérer mais cela a un coût en ressources. D'où l'idée de brider le "gratuit" à un volume modeste.\*

## Abonnements Mensuels/Annuels et Offres Entreprises

Pour les professionnels qui organisent des concours fréquents, nous proposerons des **formules d'abonnement**. L'objectif est double : **fidéliser** la clientèle avec un paiement récurrent, et offrir aux clients intensifs une solution plus économique à l'usage (coût par tirage réduit).

### Formules d'Abonnement

- **Abonnement Mensuel Standard** : par ex. *XX €/mois* pour tirages illimités (usage équitable) avec un seul compte de réseau social. Ce plan permet de faire autant de tirages que souhaité, dans la limite d'un volume de participants "raisonnable" par tirage (ex :  $\leq 40k$  commentaires par tirage, ce qui couvre 99% des cas). En cas de dépassement exceptionnel, on pourra demander de passer à l'offre supérieure. Un abonnement mensuel peut être fixé autour de ~15-20€/mois (à affiner). *Exemple réel* : Simpliers ~19,90 \$/mois pour jusqu'à 40k entrées/tirage <sup>9</sup> <sup>10</sup> ; Osortoo ~17,99 \$/mois pour illimité de commentaires <sup>11</sup>. Notre prix se situera dans cette fourchette pour rester compétitif.

- **Comptes de réseaux sociaux inclus** : par défaut, l'abonnement standard est *lié à un seul compte Instagram/Facebook/TikTok*. C'est-à-dire que l'entreprise abonnée utilisera ce plan pour tirer au sort sur son propre profil social. Nous limiterons l'utilisation à un compte *connecté* à la fois pour éviter qu'une agence abonnée pas chère n'en fasse profiter de multiples clients différents. (*Techniquement, l'utilisateur pourra lier son compte Instagram Pro, éventuellement le changer s'il le souhaite, mais pas abuser en changeant constamment.*)
- **Ajout de comptes supplémentaires** : si une même entreprise gère plusieurs comptes sociaux (par ex. une société mère avec 3 marques), on pourra autoriser l'ajout d'un 2<sup>e</sup> ou 3<sup>e</sup>

compte social dans l'abonnement pour un coût additionnel modique (ex : +5€/mois par compte). Simpliers a une approche similaire (~3,90 \$ pour un compte en plus) <sup>12</sup>. Cela couvre le cas de figure où un *même client* veut centraliser ses tirages sur plusieurs pages Instagram/Facebook sans multiplier les abonnements séparés. En revanche, cela n'est pas pensé pour un community manager gérant des marques sans lien entre elles (voir plus loin).

- **Annulation** : l'abonnement est sans engagement de durée (annulable à tout moment, arrête au prochain mois) <sup>13</sup>.

• **Abonnement Annuel (Premium)** : paiement en une fois pour 12 mois à tarif réduit (environ ~2 mois offerts, soit ~15-20% de réduction). Ex : *XX €/mois équivalent, facturé YY€ par an*. Cette formule intéressera les structures prévoyant un usage sur le long terme.

- En récompense de l'engagement annuel, ce plan **pourra inclure plus d'avantages** : par ex. *comptes sociaux illimités* (Simpliers offre les comptes illimités sur le plan annuel <sup>14</sup>), ou un volume de participants par tirage étendu (ex: jusqu'à 100k commentaires par tirage). L'idée est d'en faire la "meilleure offre" pour les pros, tout en nous assurant une trésorerie annuelle stable.
- Support prioritaire : éventuellement, les clients annuels pourraient bénéficier d'un support client prioritaire ou de fonctionnalités premium en avant-première.

• **Forfait Court-terme (Pass 48h)** : option éventuellement inspirée par Osortoo qui propose un *pass 48h à 12,99\$* <sup>15</sup>. Cela permet à un utilisateur d'accéder à toutes les fonctionnalités pendant 2 jours sans abonnement. Ce pass pourrait convenir à un **événement ponctuel** (par ex. un salon où la marque fait plusieurs tirages en un week-end). On n'avait pas évoqué initialement cette formule, mais c'est à considérer : chez nous, l'achat de crédits multiples couvre un peu ce besoin, toutefois un pass temporel en illimité 48h pourrait être une alternative. Si on l'implémente, il faudra bien le positionner par rapport aux packs de crédits pour qu'il y ait un intérêt (ex : si un client veut faire >5 tirages en 2 jours, le pass serait rentable par rapport à 5 crédits). C'est un cas d'usage moins fréquent, à prioriser en fonction de la demande.

## Offres "Entreprise" et Utilisation Multi-Comptes

Un point crucial soulevé est la **prévention des abus** : on veut éviter qu'un community manager freelance, par exemple, prenne un abonnement illimité et l'utilise pour tirer au sort les concours de *10 marques clientes différentes*, en ne payant qu'un abonnement pour potentiellement des dizaines de tirages qui auraient dû générer plus de revenus. Cela nous ferait perdre beaucoup de crédits potentiels.

### Mesures envisagées :

- Lier l'abonnement standard à un usage *mono-entreprise*. Concrètement, lors de la création du compte, l'utilisateur indique son **entreprise/marque**. Le compte pourra être vérifié (via l'email de domaine par ex.).
- Limiter le nombre de comptes de réseaux sociaux connectés sur un abonnement standard (comme vu, 1 inclus, optionnellement 2-3 max avec supplément). Ainsi, si un community manager gère 5 marques distinctes, il ne pourra pas toutes les opérer sous un seul abonnement à moins de payer des extras qui, on l'espère, le dissuaderont par rapport à prendre plusieurs abonnements ou packs de crédits séparés.
- Surveiller l'utilisation anormale : si on détecte qu'un compte utilisateur réalise des tirages pour des pages/profils très divergents souvent (ex : cette semaine un tirage pour @MarqueA, la suivante pour @MarqueB sans lien), on pourra **contacter l'utilisateur** et lui proposer une offre entreprise adaptée à son multi-client,

ou bloquer l'usage multi-profil hors de la politique. Les CGU préciseront qu'un abonnement est nominatif et réservé à une entité.

Pour les agences ou utilisateurs multi-marques légitimes, nous proposerons des **offres entreprise** :

- **Pack multi-marques** : par exemple, un abonnement mensuel spécifique où jusqu'à 5 comptes Instagram/Facebook peuvent être connectés simultanément, avec des identifiants utilisateurs distincts si besoin (pour qu'une équipe accède au dashboard). Le tarif serait évidemment supérieur à l'abonnement standard mais inférieur à 5 abonnements séparés (avantage de volume).
- **Abonnement "Agence" illimité** : pour de grosses agences, un forfait premium permettant un nombre *illimité de comptes de réseaux sociaux* et de tirages, possiblement avec un tableau de bord global et sous-comptes pour chaque marque. Ce serait notre version "SaaS business" complète. **Note**: ce genre de formule complexe peut venir dans un second temps, une fois le produit établi. Au démarrage, on peut gérer au cas par cas (sur demande) ces besoins entreprise.

#### Résumé des offres (exemple) :

- *Sans compte*: 1 crédit (1 tirage) – ~1,99€ à 2,99€.
- *Packs*: 5 crédits (~8€), 10 crédits (~15€), 20 crédits (~28€)...
- *Abonnement mensuel*: ~19,99€/mois – tirages illimités, 1 compte social (option +5€ par compte supp).
- *Abonnement annuel*: ~199€/an (équival. ~16,6€/mois) – tirages illimités, comptes sociaux illimités, fonctionnalités premium.
- *Entreprise/Agence*: sur devis ou formule dédiée (ex: 49€/mois pour 5 comptes).

Toutes ces valeurs seront ajustées lors du business plan final en comparant la concurrence et en s'assurant que c'est **rentable** pour nous (il faudra intégrer les coûts d'API payantes, voir section suivante, et les coûts serveurs).

## Intégration des Réseaux Sociaux (Instagram, Facebook, TikTok, X, ...)

L'application devra interagir avec différentes plateformes sociales pour extraire les participants aux concours (commentaires, likes, etc.) et vérifier certaines conditions. Chaque plateforme a ses **API officielles** et ses limitations, qu'il convient d'utiliser en priorité pour fiabiliser l'outil.

### Réseaux supportés en priorité

D'après les besoins exprimés, les plateformes visées pour le lancement sont :

- **Instagram** – le canal privilégié (beaucoup de concours s'y déroulent).
- **Facebook** – souvent utilisé conjointement avec Instagram (concours multi-plateformes IG + FB).
- **Twitter (X)** – bien que moins central pour les concours grand public, c'est mentionné et on le supportera pour élargir l'audience.
- **TikTok** – en forte croissance, pertinent pour cibler un public jeune. Peu de solutions actuelles supportent TikTok, ce serait un avantage compétitif.

Plateformes envisagées plus tard ou en second plan :

- **YouTube** – des concours existent via les commentaires YouTube, mais c'est moins courant pour les marques hors secteur gaming/tech. De plus, les API YouTube existent mais il faut les quotas. On peut le

garder en option à moyen terme (Simpliers et Osortoo supportent YouTube <sup>16</sup> <sup>17</sup> , donc c'est faisable).

- **LinkedIn** – réseau pro, concours rares et API complexe, probablement pas pertinent en phase initiale. On peut l'ignorer pour MVP.

- **Autres** : Pinterest n'a pas vraiment de notion de commentaires publics de concours, Bluesky est émergent (AppSorteos commence à l'ajouter <sup>18</sup> ), mais ce sera pour plus tard si la demande existe.

## Utilisation des APIs Officielles vs Scraping

### Priorité 1 : APIs officielles des plateformes

Nous privilégions l'utilisation des *endpoints* officiels (Graph API de Meta pour Instagram/FB, API Twitter, API TikTok si disponible) car :

- Cela garantit d'être dans les limites d'utilisation autorisées, - L'accès aux données est complet et fiable (ex : récupérer *tous* les commentaires via les serveurs officiels plutôt que seulement ceux chargés sur une page web) <sup>8</sup> ,

- On peut effectuer certaines actions avancées (ex: vérifier si un utilisateur suit un compte, possible via l'API dans certains cas <sup>19</sup> ).

- Cela donne de la **crédibilité** à l'outil : on peut communiquer que nous utilisons les méthodes approuvées par les plateformes (respect des CGU, pas de méthodes douteuses).

**Instagram/Facebook (Meta)** : On utilisera l'**API Graph Instagram**. Toutefois, pour pouvoir lire les commentaires d'une publication et la liste des likes, etc., l'**utilisateur doit avoir un compte Instagram Professionnel** connecté à une Page Facebook, et nous donner les permissions nécessaires (Instagram Graph API exige un token authentifié). Donc, dans l'expérience utilisateur, s'il veut faire un tirage Instagram, on le guidera pour **connecter son compte Instagram Business** via OAuth (Facebook Login). Une fois fait, on pourra appeler `/media/{media-id}/comments` pour lister tous les commentaires, avec pagination jusqu'au dernier, etc. Simpliers met en avant cette capacité de récupérer tous les commentaires directement depuis les serveurs IG <sup>8</sup> afin de garantir l'équité, et nous ferons de même.

- **Likes Instagram** : Si le concours porte sur les likes au lieu des commentaires (certains concours demandent juste de liker le post et suivre), on vérifiera la possibilité via l'API Graph. Il existe probablement un endpoint pour obtenir la liste des utilisateurs ayant liké un media. On confirmera dans la doc IG Graph. Sinon, on pourrait demander un export manuel, mais c'est moins probable, donc on se concentre sur les commentaires (cas le plus fréquent).

- **Facebook** : via l'API Graph également (les concours FB impliquent souvent de commenter une publication). Conditions similaires : la Page FB du concours doit être gérée par l'utilisateur et connectée. On récupérera les commentaires via `/post-id/comments`.

- **Limites** : L'API Graph a des quotas (appels par heure) et potentiellement des limitations sur la profondeur des commentaires (mais comme c'est le propriétaire qui accède à ses propres posts, on devrait pouvoir tout récupérer). On devra gérer les quotas et le caching intelligemment si un post a énormément de commentaires (ex: 500k). Dans ce cas, on préviendra l'utilisateur que l'opération prend plus de temps (~2000 commentaires/minute d'après Simpliers <sup>20</sup> ).

**Twitter (X)** : L'API Twitter est notoirement plus compliquée depuis 2023 (modèle payant pour accéder aux tweets et commentaires). On distingue deux usages possibles :

- **Tirage sur retweets** : beaucoup de concours Twitter demandent de RT un tweet pour participer. On pourrait alors récupérer la liste des retweets d'un tweet via l'API (ce qui nécessite un niveau d'API élevé et potentiellement payant).

- **Tirage sur réponses (commentaires)** : d'autres concours demandent de répondre au tweet avec quelque



chose. Là aussi, il faut lister les réponses (mentions) du tweet.

- **API Officielle vs Scraping** : Twitter a supprimé tout accès gratuit pour ces données. On devra soit **s'abonner à un plan API payant** (coût à intégrer dans le business plan), soit recourir à un *scraping* du tweet (via un outil interne ou un service qui parse le HTML des réponses). Le scraping est en dernier recours car non officiel et sujet aux changements. Étant donné la volonté de rentabilité, on pourrait démarrer avec une **prise en charge basique de Twitter** – ex: import manuel des participants (coller les noms) – ou utiliser une librairie existante. On surveillera l'offre de Twitter (il se peut qu'un accès *Basic* à \$100/mois soit suffisant).
- Si l'API Twitter est trop restrictive, on peut choisir de **reporter le support complet de X** ou de le limiter (par ex., l'appli pourrait permettre à l'utilisateur de télécharger ses données de tweet ou coller un export). Cependant, puisque c'est mentionné en priorité, on fera au mieux pour l'intégrer.

**TikTok** : TikTok offre une API officielle via le **TikTok for Developers**, mais c'est orienté vidéo, login, etc. Récupérer les commentaires d'une vidéo TikTok n'est pas trivial car TikTok n'a pas ouvert d'API publique simple pour cela (à vérifier s'il y a des endpoints via leur API Business maybe). On devra peut-être utiliser des solutions non officielles : soit un scraping de la page web TikTok (ce qui est faisable via un headless browser, en paginant les commentaires), soit via un service tiers. Il existe des librairies open-source qui extraient les commentaires TikTok, on peut s'appuyer dessus. **Important** : mentionner que *si* on doit recourir à ces techniques, on prendra en compte le coût en infrastructure (faire tourner un robot qui scrolle les commentaires). Peut-être que "l'IA" évoquée dans la demande fait référence à un script intelligent qui scrapera – on utilisera plus sûrement des scripts spécialisés qu'une IA au sens strict.

- **Approche** : L'utilisateur fournirait l'URL de la vidéo TikTok du concours. Le backend tenterait d'appeler l'API (si disponible à l'avenir) ou lancerait un module de scraping pour collecter les commentaires (avec limite de temps). On informera que cela peut être plus long.

- Au lancement, on peut supporter TikTok en "beta" avec mise en garde : certaines restrictions peuvent s'appliquer (ex: on ne pourra récupérer que les X premiers milliers de commentaires si non authentifié).

**Scraping en dernier recours** : Si aucune API n'est disponible ou suffisante pour un réseau, l'application utilisera *en coulisse* une méthode de récupération alternative. Cela peut passer par des appels non publics aux endpoints mobiles (certains outils utilisent les API privées Instagram, ce qui est risqué), ou par de l'automatisation headless. Par exemple, pour Instagram si un utilisateur n'a pas de compte pro (Graph API impossible), on pourrait tenter de charger la page web publique du post et de défiler pour obtenir les commentaires – mais Instagram ne charge publiquement qu'un certain nombre de commentaires (d'où la limitation de 150 souvent constatée sans login). On évitera de trop compter là-dessus pour Instagram, d'où l'importance d'inciter les utilisateurs à utiliser les **comptes pro/Business** (on pourra l'expliquer dans l'UI, cf. "conseils pour récupérer tous les commentaires" comme fait Simpliers <sup>21</sup>).

### Coûts des API et limitations financières :

Lors de l'établissement du modèle économique, on prend en compte :

- L'API Instagram Graph et Facebook n'a pas de coût direct par appel, mais a des quotas – cependant c'est globalement gratuit, on paye juste nos serveurs.

- L'API Twitter/X pourrait nécessiter un **abonnement payant** (ex: niveau Basic à 100\$/mois, ou Elevated si disponible). Ce coût fixe devra être couvert par les revenus. Si peu d'utilisateurs utilisent Twitter, on peut absorber. Sinon, on ajustera potentiellement les prix ou proposer un supplément pour Twitter (peu probable qu'on segmente, mieux vaut intégrer dans le prix global).

- L'API TikTok, si on utilise un service externe ou un scraping, c'est du temps machine de notre côté. À nous d'optimiser pour que même de gros tirages ne coûtent pas plus qu'un crédit vendu. Par exemple, on peut limiter le *refresh* des commentaires ou utiliser du cache.

- Si une plateforme facture à l'appel (ex: certaines APIs tierces le font), on réfléchira à peut-être limiter certains usages aux abonnés premium.

### Résumé intégration :

- L'application va guider l'utilisateur pour **connecter ses comptes** sur les plateformes supportées (surtout Meta). Une fois connectés, il pourra sélectionner la publication de son choix (nous pourrions soit lui demander l'URL du post, soit lister ses derniers posts via l'API – AppSorteos affiche la liste des posts du compte connecté <sup>22</sup> ). Le plus simple MVP est de demander l'URL du post et d'utiliser l'API en arrière-plan.
- On privilégiera les **serveurs officiels** pour fiabilité et complétude <sup>8</sup> . Par défaut, tous les commentaires/ participants seront pris en compte, pas seulement un échantillon (message de confiance que l'on pourra mettre en avant, cf. Simpliers qui critique les outils "bon marché, peu fiables" qui ne prennent pas tout <sup>23</sup> ).
- En **dernier recours**, pour ne pas décevoir l'utilisateur, si l'API échoue ou n'est pas dispo (ex: TikTok public), on fera du *scraping* automatique du post via l'URL fournie. L'expérience sera transparente (il collera l'URL, cliquera "Récupérer les participants", et en coulisse ça scrape). On informera juste si ça peut prendre un peu plus de temps ou s'il y a une limitation.
- **Maintenance** : Scraper les sites peut casser si les sites changent de structure. On devra donc surveiller et mettre à jour si nécessaire, ou essayer d'utiliser des API stables le plus possible.

## Critères de Participation et Filtrages Avancés

Lors d'un tirage au sort, surtout sur Instagram, il y a souvent des **règles spécifiques** que l'outil doit prendre en compte pour sélectionner éligiblement les gagnants. Notre application inclura des filtres et vérifications pour couvrir ces besoins :

### Filtrer les Participants selon les Règles du Concours

Chaque concours peut avoir des conditions du type : - « Être abonné au compte X (et éventuellement à tel partenaire) », - « Avoir liké la publication », - « Identifier 2 amis en commentaire », - « Partager la publication en story », - « Commenter X fois maximum », etc.

Notre interface de tirage permettra de **configurer ces critères** avant de lancer le tirage, et le système appliquera des filtres appropriés sur la liste des participants importés.

### Exemples de critères configurables :

- **Exclure les doublons exacts (commentaires identiques)** : Par défaut, on considère que chaque commentaire est une entrée. Mais on peut vouloir éviter qu'un même participant poste 100 commentaires identiques. On aura une option pour filtrer les doublons *stricts*.
- **Nombre de personnes mentionnées requis par commentaire** : ex: "identifier 2 personnes minimum" – on ne garde que les commentaires qui contiennent au moins 2 mentions (@). On pourra paramétrer ce minimum (1, 2, 3... ou "pas de minimum" si pas requis). *Exemple : Osortoo propose un filtre sur le nombre minimum de mentions* <sup>24</sup> .
- **Hashtag ou mot-clé requis** : si le concours demande de mettre un mot-clé dans le commentaire, on pourra activer un filtre textuel (ne garder que les commentaires contenant tel mot/hashtag). (*Osortoo a un filtre par mots-clés*) <sup>25</sup> .
- **Plusieurs entrées par personne autorisées ou non** : On donnera la possibilité soit de *limiter le nombre de participations par personne*, soit de *fusionner* toutes les entrées d'une personne en une seule. - *Mode "1 participation par personne"* : on ignore les commentaires supplémentaires d'un même utilisateur (comme si

on dédoublonnait par utilisateur). C'est utile si la règle du concours dit "une seule participation par personne" ou simplement pour équilibrer même si ce n'était pas explicite.

- *Mode "N participations max"* : au lieu de tout ou rien, on pourra choisir par ex. "max 5 commentaires par personne". Si quelqu'un a commenté 20 fois, on ne gardera que 5 de ses commentaires (les 5 premiers par exemple) dans le tirage. Ceci offre un juste milieu, très demandé par les organisateurs pour éviter que quelques participants spamment des centaines de commentaires et biaisent le hasard. - Techniquement, on peut l'implémenter en marquant chaque utilisateur et en tronquant sa liste de commentaires au nombre choisi. *Simpliers* offre un filtrage de ce genre via "Filter Unique Users" (équivalent à 1 par personne) et "Filter Unique Tags by user" <sup>26</sup> . Le second signifie qu'un même utilisateur qui tague toujours la même personne ne sera compté qu'une fois – c'est intéressant pour empêcher de tricher en mentionnant un complice en boucle. On pourra s'en inspirer.

- **Vérification de l'abonnement aux comptes requis** : Souvent, il faut *suivre le compte organisateur et éventuellement d'autres comptes partenaires*. Notre outil proposera de saisir les @ des comptes à suivre obligatoirement. Lors du tirage, une fois les gagnants tirés aléatoirement, l'outil vérifiera pour chacun s'ils suivent bien ces comptes. Comment implémenter cela ? Si l'organisateur a connecté son compte Instagram, on peut utiliser l'API pour vérifier *a posteriori* chaque gagnant : - Via l'API Graph, il n'y a pas un appel direct "user X suit-il Y ?" pour Instagram. Mais on peut potentiellement requêter la liste des followers du compte Y et chercher X. Si Y = le compte de l'organisateur (que nous avons car il est connecté), nous pouvons essayer d'obtenir son audience. Malheureusement, l'API Instagram n'offre pas directement la liste des followers pour un compte business (pour des raisons de vie privée).

- Alternative : utiliser l'API *Instagram Basic Display* (qui donne la liste des followings pour l'utilisateur authentifié, pas utile ici), ou réaliser une vérification *manuelle* rapide. Par exemple, on pourrait ouvrir le profil du gagnant via un web-scraping et voir s'il suit tel compte (sur Instagram web on peut chercher dans la liste des "following" du gagnant, mais si elle est publique seulement). Ce n'est pas garanti.

- Une façon plus pratique : **demande les preuves aux gagnants** – mais on a dit pas d'assistance humaine... Un compromis : notre outil peut **identifier le gagnant, et avant de l'annoncer, vérifier son éligibilité**. S'il manque l'abonnement, on peut soit *le disqualifier automatiquement et tirer un autre nom* (et éventuellement mentionner qu'il a été écarté pour non-respect des règles). Cela nécessite de pouvoir vérifier automatiquement.

- On peut possiblement exploiter l'**API Facebook Graph** pour ça aussi : si le concours requiert de suivre un compte Facebook, là on peut peut-être vérifier via la page likes (plus facile car les pages ont la liste de personnes qui aiment la page visible en admin). Pour IG, c'est plus fermé.

- Dans l'immédiat, on pourrait au moins indiquer dans l'interface quels gagnants tirés ne remplissent pas toutes les conditions, pour que l'organisateur puisse en choisir d'autres. *Simpliers* semble avoir une fonction "Vérifiez les gagnants avec des règles avancées" mentionnée <sup>27</sup> , donc ils ont sans doute un mécanisme similaire de post-filtrage.

- **Partage en story du post** : C'est le point le plus complexe. Le concours demande souvent "+1 participation si vous partagez ce post en story en mentionnant @nous". Vérifier cela plusieurs jours après est très difficile de manière automatisée. Comme évoqué, une méthode manuelle est de demander aux participants de taguer le compte, ainsi le compte reçoit une notification et peut faire une capture écran <sup>28</sup> . Notre but est d'automatiser et éviter la triche (quelqu'un pourrait prétendre qu'un ami a partagé mais pas le prouver, etc.).

- **Possibilité technique via API ?** Instagram Graph API ne donne pas directement la liste des stories qui mentionnent le compte. Toutefois, il existe un endpoint pour récupérer les **Mentions** d'un compte pro (càd les médias d'autres comptes où le compte est taggué). Il est possible que les stories où le compte est mentionné puissent être listées via `/`

`mentioned_media` . À vérifier. S'il est possible de récupérer les stories *actives* mentionnant le compte, on pourrait stocker ces occurrences. Par exemple, pendant la durée du concours, on pourrait régulièrement appeler l'API pour lister les nouvelles mentions en story et enregistrer qui a mentionné.

- **Limite de temps** : les stories disparaissent après 24h. Donc si le tirage a lieu après, l'info ne sera plus accessible via API (sauf à ce que l'utilisateur ait une archive, ce qui n'est pas public). Donc, pour que ce soit fiable, notre appli devrait être connectée au compte organisateur **pendant le concours** et *écouter en temps réel* les mentions. On peut imaginer de demander à l'utilisateur d'appuyer sur un bouton "Collecter les partages story" régulièrement ou de le faire toutes les X heures automatiquement sur notre serveur. On stockerait les usernames qui ont mentionné le compte. Au moment du tirage, ces usernames bénéficient d'une participation bonus.
- **Mise en œuvre** : On ajoutera une option "Story share = participation bonus". Si activée, l'interface dira "Assurez-vous d'avoir connecté votre compte Instagram et que les participants vous mentionnent. Nous suivrons les mentions." Techniquement, ça nécessite un background job. Si l'infra ne permet pas du real-time, on peut au pire demander à l'utilisateur de cliquer sur "Importer les partages story" avant de faire le tirage (ce qui fera un appel API mentions juste avant le tirage pour les 24h récentes, mais on manquera celles plus anciennes). Ce n'est pas parfait.
- **Tricherie possible** : Un CM malhonnête pourrait ajouter manuellement une liste de gens "qui ont partagé" s'il voulait favoriser quelqu'un, mais sans preuve. Pour éviter ça, on n'aura pas d'ajout manuel – c'est l'automatisation ou rien. Donc, si on ne parvient pas à garantir de capter tous les partages, on peut inclure un disclaimer que ce critère est *difficile à vérifier de façon exhaustive* et recommander d'autres critères plus objectifs. On peut toutefois *honorer le bonus story pour ceux qu'on a détectés*. Mieux vaut partiellement que pas du tout.
- Étant donné la complexité, cette fonctionnalité sera peut-être en **Beta** lors du lancement. On le signalera, et on la peaufinera avec le temps. L'important est d'être transparent : l'outil ne doit pas promettre 100% de détection si ce n'est pas sûr. (Un fil de discussion Reddit suggère aussi que beaucoup pensent que ce critère est une fumisterie invérifiable <sup>29</sup> – notre app pourrait être novatrice en proposant une solution, mais il faut être honnête sur les limites actuelles de l'automatisation).

## Détection de Comptes Doubles ou Fraude

Le problème des "concouristes" ayant plusieurs comptes pour multiplier leurs chances est bien connu. Malheureusement, du point de vue technique et légal, **il est quasiment impossible de lier deux comptes Instagram distincts à la même personne** (Instagram ne divulgue pas l'email ou le téléphone associés à un compte pour des tiers). Nous ne pouvons donc pas directement vérifier si, par exemple, `@delphine_jeuconcours` et `@delphine_jeuconcours2` appartiennent à la même personne.

### Ce que l'on peut faire :

- **Détection basique par nom d'utilisateur** : repérer des schémas évidents (ex: noms très similaires, comme X et X2). On pourrait envisager un signalement "ces comptes semblent liés". Mais ce serait hasardeux et pourrait générer des faux positifs (deux sœurs avec presque le même pseudo, etc.). Cette fonctionnalité, si demandée, serait très expérimentale.
- **Blacklist d'utilisateurs** : permettre à l'organisateur de fournir une liste de comptes à exclure du tirage. Par exemple, s'il connaît des comptes suspects (concouristes professionnels, doublons repérés), il peut les ajouter à une liste noire. Notre outil les filtrera. *Simplifiers offre la création d'une blacklist utilisateur* <sup>30</sup> . Ceci

donne un contrôle manuel au client pour contrer la fraude, sans que nous ayons à l'identifier nous-mêmes de façon incertaine.

- **Limitation via critères existants** : Les filtres précédents aident indirectement. Ex: si on exige que chaque commentaire mentionne 2 amis réels, l'utilisation de son propre 2<sup>e</sup> compte comme "ami" est détectable si on repère que deux comptes se mentionnent mutuellement beaucoup. On pourrait dans le futur avoir une alerte "ces deux comptes se tagguent entre eux de façon récurrente". Mais ce niveau de détection va au-delà du MVP.

- **Éducation** : On pourra suggérer aux organisateurs dans nos guides de règles de concours à adopter, par ex. "Évitez d'autoriser plusieurs comptes par personne, précisez dans les règles que c'est interdit". Le reste relève de leur modération.

En résumé, **on ne promettra pas la détection automatique des comptes secondaires**, car ce serait en violation des politiques (on ne peut pas demander aux participants leurs emails ou données personnelles). On fournira des outils (blacklist, filtres, tri des doublons) pour atténuer les abus, mais l'organisateur devra rester vigilant sur ce point.

## Affichage du Tirage au Sort et Expérience Utilisateur

### Format et Animation du Tirage

Pour renforcer la confiance et l'engagement, nous souhaitons que le tirage au sort soit **visuellement démonstratif**. L'idée est de pouvoir partager une animation type *Story Instagram* ou *vidéo TikTok* montrant le gagnant sélectionné aléatoirement.

#### Fonctionnalités d'affichage :

- **Déroulement animé des participants** : Une fois que la liste des participants éligibles est prête (après application des filtres), le lancement du tirage affichera une animation ludique. Par exemple : les noms ou pseudos des participants défilent rapidement à l'écran, éventuellement accompagnés de leur photo de profil. Cela peut être présenté comme une roue qui tourne, un slot machine, ou simplement une succession rapide de cartes avec les noms, puis ralentit et s'arrête sur le gagnant. Ce procédé visuel rappelle les tirages télévisés et prouve qu'il y a bien un élément aléatoire. *C'est un standard des outils modernes : Simplifiers affiche même en temps réel les commentaires qui défilent pendant la collecte* <sup>31</sup>, et d'autres apps permettent d'enregistrer la vidéo du tirage.

- **Format Story 9:16** : On priorise un format vertical (1080x1920 par ex) qui correspond aux Stories Instagram et aux vidéos TikTok, pour que l'organisateur puisse directement partager le résultat en story. L'interface mobile de notre app affichera donc le tirage en plein écran vertical. Sur web, on peut avoir un mode "aperçu story" ou un format horizontal classique selon l'usage, mais on veillera que l'animation puisse être exportée en vertical.

- **Personnalisation minimale** : Au départ, ce sera un thème générique (fond neutre ou légèrement brandé avec notre logo par ex., confettis animés pour la célébration du gagnant). Pas de custom aux couleurs de la marque pour le MVP, afin de rester passe-partout. L'important est que ce soit *moderne et propre visuellement*, sans en faire trop. *Inspiration : les interfaces épurées type Revolut/Wise pour la fluidité*. On évitera les cliparts de cadeau kitsch, pour un look plus *flat design* contemporain.

- **Annonce du gagnant** : Une fois la sélection faite, l'écran affiche clairement le *nom du gagnant* (et éventuellement son avatar) avec une mention du type " Félicitations @pseudo, tu as gagné ! ". S'il y a plusieurs gagnants, ils s'affichent l'un après l'autre ou ensemble selon l'option (on peut imaginer tirer 3 gagnants en même temps avec 3 cartes animées distinctes). Pour un tirage avec suppléants, on peut

afficher "Gagnant : X, Suppléants : Y et Z".

- **Enregistrement vidéo** : L'application proposera d'**enregistrer l'animation sous forme de vidéo** (MP4) ou GIF une fois le tirage terminé <sup>32</sup>. Ainsi, l'utilisateur peut sauvegarder cette preuve et la poster sur ses réseaux. Sur mobile, on pourra offrir un bouton "Partager en story" qui génère directement la story (via l'API de partage IG par ex). Sur web, l'utilisateur pourra télécharger la vidéo ou la QR code sur son téléphone pour la publier.

**Exemple d'animation simple** : un écran avec le logo du concours (optionnel), puis un carrousel des noms qui défilent. On peut avoir un compteur ou un *compte à rebours* de 3 secondes visible, puis la sélection freeze. Ce *compte à rebours personnalisable* est même une feature qu'on peut ajouter (Simpliers permet de régler la durée de l'animation <sup>33</sup>, par défaut 8 secondes). Pour la v1 on peut juste faire ~5 secondes par exemple, éventuellement paramétrable plus tard.

**Pas de surcharge d'effets** : On évite les animations trop longues ou complexes (pas besoin d'une 3D extravagante). L'UX doit rester **rapide** : l'utilisateur clique "Tirer au sort" et en quelques secondes le résultat est connu, tout en ayant un minimum de show visuel.

## Interface Utilisateur : Parcours Simplifié

On visera une interface **très simple d'utilisation** malgré la richesse des fonctionnalités. Voici un parcours utilisateur type :

1. **Accueil / Tableau de bord** : si connecté, l'utilisateur voit un dashboard avec ses derniers tirages et un bouton "Nouveau Tirage". S'il n'est pas connecté (usage one-shot), il voit directement un champ pour entrer l'URL du post et lancer un tirage (avec indication du coût).
2. **Nouvelle Opération de Tirage** : l'utilisateur renseigne l'URL du post du concours (ou choisit parmi ses posts si son compte social est lié). Il sélectionne la plateforme (IG, FB, etc.) si ce n'est pas évident via l'URL.
3. Si le compte n'est pas encore connecté et que l'API le nécessite, on lui demandera de connecter son compte (ex: "Connectez votre Instagram pour récupérer tous les commentaires"). Sinon, on procédera en mode public/scraper avec avertissement si applicable.
4. **Chargement des participants** : l'application récupère les commentaires/participants. Une barre de progression ou animation en temps réel montre l'avancée (ex: "Chargement de 1200 commentaires..."). Pendant ce temps, on peut afficher quelques messages sur l'équité du processus, etc., ou de la pub pour nos offres.
5. **Paramétrage des règles** : une fois les données prêtes, l'UI affiche le nombre total d'entrées trouvées et propose les filtres : ex: "Exclure doublons par utilisateur ? (oui/non)", "Max commentaires par personne: [ 1 / 5 / 10 / illimité ]", "Nb minimum de mentions dans le commentaire: [0/1/2...]", "Doit contenir hashtag ? : [champ texte]", "Comptes à suivre obligatoirement: [@compte1, @compte2]", "Bonus story activé ? (oui/non)". Ces options sont claires et par défaut réglées sur la configuration la plus permissive (pour ne pas exclure involontairement). L'utilisateur peut lire des info-bulles d'aide pour comprendre chaque filtre.
6. On peut aussi offrir un bouton "Règles IA" où l'IA remplirait automatiquement les critères en analysant le texte du post (Simpliers a un auto-fill AI des règles <sup>34</sup>), mais ce serait bonus plus tard.
7. **Tirage** : l'utilisateur clique "Tirer le(s) gagnant(s)" après avoir précisé le nombre de gagnants à sélectionner et éventuellement le nombre de suppléants. Le crédit est alors consommé (si besoin, on

vérifie qu'il en a un – sinon il est invité à en acheter à ce moment). L'animation se lance comme décrit plus haut, puis affiche le résultat.

8. **Résultat** : Le(s) gagnant(s) sont listés. L'interface propose : "Enregistrer la vidéo du tirage", "Copier le résultat dans le presse-papier" (pour coller dans une légende Instagram par ex), "Générer un certificat de validité" (PDF ou lien public), et "Publier les gagnants sur [Instagram Story/Facebook]" via un partage direct.
9. Si la vérification de suivi était activée, on peut ici signaler par un petit badge vert/rouge si chaque gagnant suit bien les comptes requis. En cas de non-respect, on peut avoir un bouton "Retirer ce gagnant non-éligible et tirer un autre à sa place". Ainsi l'organisateur a la main pour garantir les règles ex-post.
10. **Sauvegarde et historique** : Le tirage réalisé s'enregistre dans l'historique du compte (si connecté). On lui donne un nom par défaut (ex: "Tirage du 5 Dec 2025 – Post Instagram 12345"). L'utilisateur peut renommer pour s'y retrouver (ex: "Concours Noël 2025 – partenariat X"). Dans l'historique, il pourra voir la liste des gagnants et suppléants, l'URL du post, la date/heure du tirage.

Ce parcours doit tenir en peu d'écrans, être **fluide**. On s'inspirera des apps grand public (Revolut, Wise, TooGoodToGo) pour la navigation intuitive, en évitant les informations superflues. Par exemple, un design minimaliste avec des icônes claires, des boutons bien espacés, et un ton convivial (sans être trop enfantin).

**Multilingue** : Dès le début, on peut prévoir l'appli en français et en anglais (au moins). Le nom potentiellement sera en anglais, mais il faut que le texte de l'UI puisse se traduire (ex: "Draw a winner" / "Tirer au sort"). On a vu que certaines apps comme Rafi ou Simpliers gèrent plusieurs langues <sup>35</sup>, c'est un plus pour toucher un public large.

## Nom de l'Application

Trouver un nom **évocateur, moderne et disponible** est une étape importante. Quelques pistes et idées :

- **Nom descriptive en anglais** : *Giveaway Picker*, *FairDraw*, *RandomWinner*, *LuckyPick*. Avantage : parle à un public international, décrit la fonction. Inconvénient : peut être peu original (plusieurs outils s'appellent "Instagram Giveaway Picker" par ex).
- **Nom en français** : *Tirage+*, *Au Hasard*, *Chanceur*, *TAS Multi*. Avantage : clair pour le public francophone cible principal, peut souligner le sérieux (français = confiance locale). Inconvénient : moins passe-partout si on vise l'international ensuite.
- **Nom mixte ou abstrait** : Un mot inventé ou contraction, facile à prononcer dans plusieurs langues. Par ex: *Pickaly*, *DrawIt*, *RaffleX*, *Hasardum* (mix hasard + random), *Randowin*. Ou un mot court réel qui évoque le hasard ou la sélection : *Plouf* (référence au "plouf plouf"), *Lotus* (pour loterie, mais connotation fleur), *Dice* (dé).

### Critères souhaités :

- Fait comprendre l'usage "tirage au sort" en un clin d'œil, ou via un sous-titre.
- Évite l'imagerie cadeau trop datée (beaucoup d'apps 2010's utilisaient "gift", on veut se démarquer).
- Possibilité de versionner/localiser le nom : par ex, un nom principal en anglais mais qui se décline en français dans le store (comme *Pregnancy+* qui devient *Grossesse+* localement). Si on trouve un terme court, ce n'est peut-être pas nécessaire.
- Disponible en nom de domaine .com/.fr idéalement, et pas déjà pris sur les stores.

### Propositions de nom :

- **FairPick** – insiste sur l'équité (fair) et pick (choisir). Simple et mémorable.
- **RanDraw** (jeu de mot entre random et draw) – évoque le tirage aléatoire.
- **Tirage+** – court et intègre le symbole plus, tendance dans les apps (réutilisable en *Draw+* pour l'anglais).
- **LuckyTAS** – mélange anglais "lucky" et TAS (tirage au sort), compréhensible FR/EN.
- **Pickify** – style moderne, sonne comme "simplifier le pick".

On devra tester ces idées, voir s'il n'y a pas de conflit. À première vue, *"FairPick"* ou *"Tirage+"* ressortent bien. On écarte les trucs trop génériques comme *"GiveawayApp"* ou trop connotés cadeaux. Une fois le nom arrêté, on créera le logo en conséquence.

*(Pour l'instant, nous continuerons de mentionner "l'application" ou "notre service" en attendant de fixer le nom.)*

## Design UI/UX et Identité Visuelle

Le design sera **minimaliste et neutre** de sorte à convenir aux utilisateurs venant de différents réseaux (Instagram a une charte plutôt colorée rose/orange, Facebook est bleu, Twitter bleu ciel, TikTok fond noir/fluos, etc.). Nous ne voulons pas trop nous aligner sur l'un ou l'autre, afin que *l'application soit perçue comme un outil impartial, générique*.

### Charte graphique :

- Couleurs dominantes : probablement un **fond clair** (blanc ou gris très pâle) pour une interface aérée. Des touches de couleur vive pour les éléments importants (bouton "Tirer", indicateurs). On peut emprunter des tonalités modernes type **bleu-vert** (teal) ou **violet** vibrant, qui évoquent la tech moderne, tout en restant sobres. Éviter le combo noir/orange fluo de TikTok ou le bleu lourd de Facebook – rester neutre.
- Thème : clair en standard. Un mode sombre pourrait être proposé plus tard, mais ce n'est pas prioritaire.
- Typographie : sans-serif moderne (proche de Helvetica/Roboto). Lisible, et cohérente entre web et mobile.
- Icônes et illustrations : des icônes linéaires simples (style Material Design icons ou Feather icons). Par exemple un icône de trophée pour représenter les gagnants, un icône de filtres, etc. Pour l'animation de tirage, on peut concevoir une petite mascotte ou animation générique (ex: une main qui mélange des papiers, un dé qui roule...). Mais attention à ne pas être kitsch. Peut-être simplement confettis + trophée suffiront.
- Layout : **très épuré**. On suivra les bonnes pratiques d'UX : pas de surcharge d'options sur un seul écran, utilisation de boutons clairs, de sliders ou switches pour activer les filtres plutôt que des checkboxes austères, etc. Des *espaces blancs* suffisants pour rendre la lecture confortable (on évite de gros pavés de texte). Par exemple, lors du paramétrage des règles, on pourrait afficher chaque option dans une carte ou une section bien délimitée, avec un petit descriptif.

### Inspiration :

- *Revolut* – pour son UX fluide, animations subtiles et interface moderne (on aimera notamment la clarté des écrans et la sensation de réactivité). On ne prendra pas son fond sombre par défaut, mais on peut s'inspirer de ses graphismes plats et de l'usage de dégradés élégants.
- *Wise* – pour la modernité et la simplicité visuelle.
- *TooGoodToGo* – pour la facilité de navigation (c'est très user-friendly, tuto de démarrage clair, etc.).



On cherchera à provoquer un **sentiment de confiance et de professionnalisme**. L'appli doit paraître assez sérieuse pour qu'une grande marque l'utilise (donc pas un design enfantin), tout en restant conviviale pour qu'organisateur et participants s'y retrouvent.

Au niveau du branding, on peut avoir un logo sobre. Pas besoin d'intégrer un motif cadeau car le client a exprimé qu'il trouve cela daté. Peut-être un symbole comme des *cartes mélangées* ou un *petit jeton aléatoire* plus abstrait.

**Expérience fluide** : l'application guidera l'utilisateur par des micro-textes explicatifs, mais limités. On veut qu'en ouvrant l'app, en quelques clics le tirage soit fait (d'où l'importance de la simplicité du parcours). On inclura éventuellement un **onboarding** (3 écrans tutoriels au premier lancement) pour montrer "1. Connecte ton compte, 2. Choisis ton post, 3. Tire au sort en 1 clic", afin que l'utilisateur comprenne la proposition de valeur immédiatement.

## Enregistrement des Résultats et Preuve du Tirage

La crédibilité de l'outil sera renforcée en permettant aux organisateurs de **prouver le bon déroulement** du tirage, notamment vis-à-vis de leur audience ou en cas de contrôle (huissier, règlement de jeu concours, etc.). Pour cela, plusieurs fonctionnalités seront intégrées :

- **Historique des tirages sur le compte** : Comme évoqué, chaque tirage effectué avec un compte connecté sera sauvegardé. Dans l'onglet "Historique" ou "Mes Tirages", on verra une liste avec : date, réseau, URL du post, intitulé (personnalisable), nombre de participants à l'époque, gagnant(s) tiré(s). En cliquant sur un tirage passé, on pourra afficher les détails : liste complète des participants pris en compte (on pourrait stocker un ID de tirage pour permettre d'en récupérer la liste exacte), les paramètres utilisés (filtres, etc.), et les gagnants. Cela sert de **journal** en cas d'audit. Par exemple, si une personne conteste en disant "mon commentaire était là mais je n'ai pas été pris en compte", l'organisateur peut vérifier dans les participants exportés. *Simplifiers permet d'exporter la liste complète des participants et la partager comme preuve* <sup>31</sup>, ce qui est une bonne pratique de transparence.
  - L'historique sera stocké sur nos serveurs de manière sécurisée et conforme RGPD (voir plus bas). On pourrait le conserver indéfiniment tant que le compte existe, ou purger au bout d'un certain temps selon contraintes de stockage. Mais comme c'est surtout du texte (pseudos), ce n'est pas énorme en volumétrie même sur des milliers de concours.
  - **Recherche dans l'historique** : Pour les comptes qui font beaucoup de tirages, on ajoutera une barre de recherche par mots (ex: retrouver "MELI MATCHA" dans l'exemple mentionné). Pouvoir renommer les tirages est important pour s'y retrouver (ex: "Concours Noël 2025 avec Machin" plutôt que juste un code).
- **Certificat de tirage** : On générera pour chaque tirage un **certificat de validation** – par exemple un PDF ou une page web publique – qui contient les informations clés : nom du concours, date/heure UTC du tirage, nombre de participants, algorithme utilisé (ex: "tirage aléatoire uniforme via générateur de nombres pseudo-aléatoires sécurisé"), identifiant unique du tirage, et le résultat (nom

des gagnants). Ce certificat pourra être **fourni au service juridique** si nécessaire, ou partagé avec l'audience pour la transparence.

- On pourra inclure un code de vérification (hash) pour prouver que le document n'a pas été altéré. Peut-être pas nécessairement une signature numérique complexe, mais au moins un checksum lié à l'ID du tirage.
  - L'application Simpliers offre une page de résultats partageable ainsi qu'un certificat <sup>36</sup> <sup>2</sup> . On fera de même : par exemple un lien du style `notretirage.com/result/ABC123` que l'organisateur peut poster pour que n'importe qui puisse voir la preuve (liste des commentaires, etc.). Cela rassure les participants sur l'intégrité du concours.
  - **Preuve vidéo** : En sus du certificat textuel, la vidéo de l'animation du tirage sert aussi de preuve car on y voit le processus se dérouler en direct. Beaucoup d'organismes publient simplement la vidéo du tirage, c'est souvent suffisant pour le public. On facilitera donc le partage de cette vidéo (comme discuté).
- **Export des données** : On permettra à l'organisateur **d'exporter la liste des participants** et le résultat (CSV/XLS). Ainsi, il peut garder dans ses archives, ou utiliser pour contacter les gagnants par mail (si on avait des mails, mais ici on n'a que des pseudos). Au moins, un export CSV des commentaires participants pourrait être utile pour un suivi. *Simpliers inclut l'export XLS des participants* <sup>37</sup> .
- **Logs internes** : Pour notre usage, on conservera un log technique du tirage (timestamp, seed aléatoire utilisé, etc.) en base de données. Ceci en cas de bug ou contestation, pour qu'on puisse rejouer ou prouver ce qui a été fait. Ce log ne sera pas forcément exposé aux utilisateurs mais peut servir au support client ou audit.

**Garantir l'équité** : dans notre communication, on insistera sur le fait que *personne, pas même nous les développeurs, ne peut truquer le résultat une fois le tirage lancé* <sup>38</sup> . Cela peut être assuré par la manière dont le code est écrit (on utilise un générateur aléatoire fiable, on ne permet pas de retirer un nom sauf via les filtres prédéfinis, etc.). Le certificat avec un hash peut aider à prouver qu'aucune modification n'a eu lieu après coup.

Enfin, pour les **grandes marques** qui ont des exigences légales (parfois huissier etc.), notre outil pourra être présenté comme conforme. Eventuellement, à terme, on pourrait envisager une **certification par huissier** de notre logiciel, ou au moins rendre l'huissier capable d'assister au tirage via l'application (par ex. en lui donnant accès à la liste initiale et au random seed). C'est du bonus, mais mentionné en point 15 ("Prevois la preuve de tirage au cas où pour les grandes marques"), donc à garder en tête.

## Conformité Légale, Sécurité et Données RGPD

Étant basé en France/UE, le service sera entièrement conforme au **RGPD** et aux lois locales encadrant les jeux-concours.

## Localisation des Données et Hébergement

- Toutes les données personnelles et données de concours collectées seront stockées sur des serveurs situés en **Europe** (par exemple centres AWS en France ou en UE). Pas de transfert hors UE sans nécessité, afin de se conformer aux directives de protection des données.
- On nommera un **DPO (Data Protection Officer)** ou référent RGPD, chargé de répondre aux demandes des utilisateurs concernant leurs données. Les CGU et la politique de confidentialité préciseront comment nous gérons les données.

## Données Collectées

- **Côté participants aux concours** : on collecte potentiellement leurs pseudos, commentaires, et informations publiques associées via les APIs (par exemple la photo de profil, ou l'ID public). Ces données sont utilisées pour le tirage et la preuve. Elles peuvent être considérées comme des données personnelles (un pseudo + photo pouvant identifier une personne). Selon le RGPD, notre client (organisateur) est en quelque sorte le responsable de traitement de la liste des participants, et nous sommes sous-traitant. Il faudra insérer dans les règles du concours une mention que l'organisateur utilise un outil tiers pour le tirage et que les données publiques des participants seront traitées dans ce cadre.
- **Durée de conservation** : on conservera l'historique des tirages aussi longtemps que le compte client existe, sauf demande de suppression. Néanmoins, on peut mettre en place une **purge** optionnelle : par ex, permettre à l'utilisateur de supprimer un ancien tirage de l'historique, ce qui effacera les données associées (liste de participants, etc.) de nos bases. Par défaut, on garde tout pour l'archive (car c'est souvent utile pour prouver en cas de litige tardif). On indiquera dans la politique qu'on conserve ces données d'historique X années (peut-être 3-5 ans) sauf demande contraire.
- **Données des comptes utilisateurs** : email, hash MP, etc., conservés tant que le compte est actif + X temps après suppression (pour permettre réactivation par ex ou obligations légales).
- **Cookies et tracking** : on évitera tout tracking superflu. Peut-être un cookie de session pour le login, analytics anonymisés éventuellement (à mentionner). RGPD oblige un bandeau cookies si analytics ou si on a un site vitrine.

## Droits des Utilisateurs

Conformément au RGPD, les utilisateurs (organisateurs) auront un droit d'accès, rectification, suppression de leurs données personnelles. Via l'interface ou support :

- **Accès** : possibilité de voir les infos du compte, de télécharger les données (par ex: l'historique de leurs tirages, leurs factures).
- **Suppression** : possibilité de supprimer leur compte. Cela supprimera ou anonymisera en base leurs infos perso. (On pourrait anonymiser les tirages passés au lieu de tout effacer, car ça concerne potentiellement les participants. Ou bien on supprime tout car ces tirages n'ont plus de raison d'être si l'utilisateur est parti). On doit respecter la demande en tout cas.
- **Portabilité** : pas vraiment applicable ici à part fournir exports des tirages (CSV déjà prévu).
- **Opposition** : ils peuvent choisir de ne pas recevoir nos emails marketing etc.

## Sécurité & Confidentialité

- On ne fera aucune utilisation commerciale des données des participants aux concours. Pas de revente de listes de pseudos, etc. C'est strictement pour le tirage. On l'explicitera pour rassurer les clients (beaucoup de marques y tiennent).
- Les communications avec les APIs tierces respecteront leurs conditions (ex: on n'enregistrera pas de jetons d'accès plus que nécessaire, etc.).
- Un point à mentionner : les concours sur réseaux sociaux sont soumis aux **règles des plateformes** (par ex Facebook/Instagram demandent de décharger la responsabilité, etc.). Notre outil facilitera le respect de ces règles, mais l'organisateur doit toujours rédiger un règlement de concours. On pourra fournir un **générateur de termes légaux** (AppSorteos propose ça <sup>39</sup>) ou un modèle type, éventuellement plus tard.

**Conclusion partie légale :** On s'alignera sur la législation franco-européenne, plus stricte que celle des US, ce qui garantit un haut niveau de protection. Toutes ces mesures seront détaillées dans nos conditions d'utilisation et politique de confidentialité, rédigées avec l'aide d'un juriste idéalement. L'objectif est d'offrir le service en toute conformité, afin que les grandes entreprises n'aient aucun frein légal à l'utiliser.

## Perspectives Techniques et Fonctionnalités Futures

*(Pour clore, listons rapidement quelques aspects techniques supplémentaires et évolutions possibles.)*

- **Backend extensible (API REST) :** On développe dès maintenant une API interne documentée, de sorte qu'à l'avenir on puisse ouvrir une partie en API publique (par ex., si un client veut intégrer le tirage à son propre back-office via notre API). En outre, ça permet de séparer front et back pour éventuellement offrir un **mode SaaS complet** plus tard où de gros clients utiliseraient juste l'API. Pour l'instant, l'accent est sur l'appli elle-même, mais cette architecture ouverte nous garde de la flexibilité.
- **Scalabilité :** Si le service monte en charge (ex: une opération publicitaire crée des milliers de tirages en même temps), on pourra passer le backend en microservices ou serverless pour absorber les pics (par ex, un service dédié pour récupérer les commentaires, un autre pour la logique de tirage, etc.). On prévoit la conception modulaire dès le départ, sans trop complexifier initialement.
- **Multiplateforme support complet :** À terme, ajouter YouTube, LinkedIn s'il y a une demande. Également intégrer peut-être des outils hors réseaux traditionnels, par ex. tirage parmi une liste d'emails (pour des newsletter giveaways) – certains concurrents le font.
- **Évolution UX/UI :** Offrir plus de personnalisation (thèmes de l'animation, possibilité d'ajouter le logo de la marque sur l'animation ou page de résultat). Sur abonnement premium, par ex, pas de mention de notre app sur la vidéo/certificat (white-label partiel) <sup>40</sup>.
- **Fonctionnalités sociales :** un annuaire public des tirages (comme AppSorteos propose un répertoire <sup>41</sup>)? Pas forcément utile, mais un argument de transparence. On peut lister les concours terminés si les marques veulent être référencées, mais la plupart préfèrent gérer ça sur leurs pages.
- **IA et automatisation :** outre l'auto-remplissage des règles, on peut imaginer utiliser l'IA pour détecter les commentaires hors-sujet, repérer des tricheries subtiles, ou pour générer un petit texte d'annonce du gagnant (ex: "Bravo @X, vous remportez le lot Y du concours Z"). Ce n'est pas prioritaire, mais dans une V2 ça peut ajouter de la valeur.

# Stratégie de Lancement

Afin d'assurer un lancement réussi de l'application, nous allons élaborer une stratégie marketing et opérationnelle :

- **Beta test privée** : recruter quelques community managers ou petites entreprises volontaires pour tester une version bêta. Ils auront des tirages gratuits en échange de leurs retours. Cela permettra de peaufiner l'UX et de s'assurer que les principales plateformes (IG en tête) fonctionnent sans accroc. On peut cibler via des groupes Facebook de social media managers, ou LinkedIn, des personnes susceptibles d'utiliser l'outil.
- **Positionnement et différenciation** : Sur le marché, il existe déjà plusieurs outils de tirage au sort (Simplifiers, AppSorteos, Osortoo, des apps mobiles comme Rafi, etc. – voir comparatif HubSpot <sup>42</sup>). Notre communication devra insister sur nos points forts : *multiplateforme complet* (IG, FB, TikTok, Twitter dès la v1, là où certains en ignorent), *tarification transparente* (pas de mauvaise surprise ni de système de crédits obscur), *UI moderne et rapidité, conformité RGPD* (hébergement UE, ce que les outils US n'offrent pas toujours), et *petits plus* comme la vérification de règles automatisée et l'animation partageable. C'est notre argumentaire de vente.
- **Marketing digital** :
  - Lancement d'un site web vitrine (**landing page**) clair, en plusieurs langues possiblement, avec démonstration du produit (GIF de l'animation, captures d'écran), explication des offres, FAQ (expliquer comment on fait pour IG, mentionner la nécessité compte pro IG, etc.), et un blog ou section ressources (conseils pour réussir un concours, etc.).
  - SEO : Se positionner sur des requêtes comme "tirage au sort Instagram", "Instagram giveaway tool", en français et anglais. On produira du contenu sur ces thèmes pour remonter dans Google. Ex : des articles type "Les 10 meilleurs outils de tirage au sort Instagram" où évidemment on se cite, ou "Comment faire un concours Instagram conforme". Également, être référencé sur des blogs de marketing digital (HubSpot, etc.).
  - **Réseaux sociaux** : quoi de mieux que d'utiliser nous-mêmes les réseaux pour promouvoir l'outil ! On peut créer une page Instagram, une page Facebook, un compte Twitter pour l'app. Sur Instagram, on pourrait organiser un *faux concours* ludique où le gagnant remporte un abonnement gratuit, en utilisant bien sûr notre propre outil pour le tirage – manière de montrer en situation réelle. Publier régulièrement des *astuces concours*, des *success stories* de clients (ex: "Regardez comment la marque X a gagné 500 abonnés grâce à un concours, tiré au sort avec [notre app]").
  - **Partenariats influenceurs** : repérer des influenceurs ou agences marketing qui pourraient recommander notre solution. Leur fournir un essai gratuit ou un code promo pour qu'ils en parlent. Par exemple, un YouTuber qui parle de social media tools pourrait faire un tuto de notre appli (ce qui en plus rassure sur comment l'utiliser).
  - **App Stores** : Optimiser le référencement sur l'App Store et Play Store si on y est : soigner la description (mots-clés "tirage concours", "Instagram", etc.), images, et recueillir vite des avis positifs (peut-être via nos beta users). Sur l'App Store, il y a déjà une app nommée "Tirage au sort Instagram" classée 4+ <sup>43</sup>, on essaiera de se distinguer peut-être par un nom plus unique.
- **Tarification de lancement** : On peut offrir un **pack de bienvenue** (ex: 3 crédits offerts à la création d'un compte) pour inciter l'inscription. Ou une **réduction de lancement** sur l'abonnement annuel

pour les premiers inscrits. Ce genre de promotion limitée crée un sentiment d'urgence et attire des clients pilotes.

- **Support client & communauté** : Au lancement, prévoir une petite équipe/support (même si c'est nous au début) disponible pour aider les nouveaux utilisateurs, par chat ou email. Créer une **FAQ en ligne** avec les questions courantes (ex: "Pourquoi je dois avoir un compte pro Instagram ?", "Comment régler tel filtre ?", etc.). Éventuellement mettre en place un petit forum ou groupe Facebook pour fédérer les utilisateurs, où ils pourront partager des suggestions.
- **Évolutions post-lancement** : Recueillir les feedbacks et rapidement corriger les bugs ou manques critiques. Montrer une roadmap publique peut être bien (ça inspire confiance de voir qu'on a des plans de dev continus). Par ex, "bientôt : personnalisation de l'animation, support de YouTube, etc."
- **Légal lancement** : S'assurer avant pub massive que tout est prêt niveau légal (CGU, mentions légales, conformité store). On aura vérifié avec un avocat les conditions pour éviter des soucis (surtout sur la gestion des données des participants).

Avec cette stratégie, on vise à **imposer notre outil comme la référence en France** d'abord, puis éventuellement en Europe, pour les tirages de concours sur réseaux sociaux. La clarté de l'offre et la robustesse technique seront nos meilleurs alliés pour convertir les utilisateurs méfiants qui utilisent encore des méthodes bricolées ou des apps douteuses.

---

*Avec ce document, toutes les informations essentielles sont réunies pour passer à la phase de développement et de mise en œuvre. Le prochain livrable pourra inclure des wireframes UI détaillés basés sur ces spécifications, afin de valider l'ergonomie avant codage. L'équipe technique pourra également estimer les efforts et commencer à planifier l'architecture logicielle en s'appuyant sur le stack et les modules décrits.*

Nous avons ainsi une vision complète du produit (**README**) qui servira de référence tout au long du projet.

19 26

---

1 3 4 5 32 35 43 Tirage au sort Instagram dans l'App Store

<https://apps.apple.com/fr/app/tirage-au-sort-instagram/id1601837783>

2 6 9 10 12 13 14 19 26 30 33 37 Simpliers - Prices

<https://simpliers.com/en/giveaway/prices>

7 11 15 16 17 24 25 Sélectionnez les gagnants sur Instagram, Facebook et YouTube - Osortoo

<https://osortoo.com/fr/>

8 20 21 23 31 34 36 38 Simpliers - Sélecteur de cadeaux Instagram gratuit

<https://simpliers.com/fr/concours/instagram>

18 22 39 41 AppSorteos – Tirage au Sort Instagram

<https://app-sorteos.com/fr/apps/tirage-au-sort-instagram>

27 40 Simpliers - Des prix

<https://simpliers.com/fr/concours/des-prix>

28 29 Comment fonctionnent les concours sur Instagram ? : r/AskFrance

[https://www.reddit.com/r/AskFrance/comments/193vfr7/comment\\_fonctionnent\\_les\\_concours\\_sur\\_instagram/](https://www.reddit.com/r/AskFrance/comments/193vfr7/comment_fonctionnent_les_concours_sur_instagram/)

42 Tirage au sort Instagram : les 6 meilleurs outils - Blog HubSpot

<https://blog.hubspot.fr/marketing/tirage-au-sort-instagram>