

PEETIC RICE × KANO

Priorisation RICE filtrée par Kano Model

2025

TABLE DES MATIÈRES

1.	Introduction aux frameworks RICE et Kano	4
1.1.	Le framework RICE	4
1.2.	Le modèle Kano	6
2.	Application RICE à Peetic	9
2.1.	Méthodologie appliquée	9
2.2.	Tableau RICE complet - Toutes solutions	10
2.3.	Analyse des scores RICE	11
3.	Filtrage Kano appliqué au RICE	12
3.1.	Méthodologie de double filtrage	12
3.2.	Classification Kano des solutions Peetic	13
4.	Roadmap priorisée RICE × Kano	15
4.1.	Phase 1 : Q1 2026 (Fondations + Quick Wins)	15
4.2.	Phase 2 : Q2 2026 (Performance + Delighter validé)	16
4.3.	Phase 3 : Q3-Q4 2026 (Densification)	17
5.	Analyse comparative RICE vs OST	18
5.1.	Convergence entre frameworks	18
6.	Matrice de décision finale RICE × Kano	19
6.1.	Tableau de synthèse	19
6.2.	Visualisation matricielle	20
7.	Métriques de succès par solution	21
7.1.	Définition des métriques RICE-aligned	21
8.	Risques et mitigation	23
8.1.	Risques identifiés par RICE	23
8.2.	Risques Kano spécifiques	24
9.	Utilisation pédagogique	25
9.1.	Exercices RICE pour formateurs	25
9.2.	Pièges à éviter	26
10.	Conclusion	27
10.1.	Synthèse RICE × Kano pour Peetic	27
10.2.	Prochaines étapes	27
11.	Références	28

Contact

PABLO PERNOT
pablo@projetwinston.fr
06 32 43 78 03

Licence

CC BY-SA 4.0
<https://creativecommons.org/licenses/by-nd/4.0/>

1. INTRODUCTION AUX FRAMEWORKS RICE ET KANO

1.1. LE FRAMEWORK RICE

1.1.1. DÉFINITION

RICE est un framework de priorisation produit développé par Intercom qui permet d'évaluer des initiatives basées sur 4 critères quantifiables :

- *R*each (Portée)
- *I*mpact
- *C*onfidence (Confiance)
- *E*ffort

1.1.2. FORMULE DU SCORE RICE

Le score RICE se calcule selon la formule :

$$\text{Score RICE} = (\text{Reach} \times \text{Impact} \times \text{Confidence}) \div \text{Effort}$$

1.1.3. DÉTAIL DES COMPOSANTES

1.1.3.1. Reach (Portée)

Nombre d'utilisateurs impactés par trimestre. Mesuré en nombre absolu.

Exemples :

- Feature utilisée par 80% des utilisateurs actifs : Reach = 800 (si 1000 MAU)
- Feature de niche touchant 5% : Reach = 50

1.1.3.2. Impact

Contribution à l'objectif produit, sur échelle normalisée :

- 3 = Impact massif
- 2 = Impact élevé
- 1 = Impact moyen
- 0.5 = Impact faible
- 0.25 = Impact minimal

1.1.3.3. Confidence (Confiance)

Certitude dans les estimations R, I, E. Exprimé en pourcentage :

- 100% = Données solides, tests validés
- 80% = Bonnes hypothèses, quelques tests
- 50% = Intuition, pas de validation
- < 50% = Pure spéculation (à éviter)

1.1.3.4. Effort

Travail nécessaire, mesuré en personne-mois.

Exemples :

- 1 dev pendant 2 semaines = 0.5 personne-mois

- 2 devs + 1 designer pendant 1 mois = 3 personne-mois

1.2. LE MODÈLE KANO

1.2.1. DÉFINITION

Développé par Noriaki Kano (années 1980), ce modèle classe les fonctionnalités selon leur impact sur la satisfaction client.

1.2.2. LES 5 CATÉGORIES KANO

1.2.2.1. 1. Basic Needs (Besoins de base / Must-haves)

Fonctionnalités **attendues** par l'utilisateur. Leur absence crée de l'insatisfaction, mais leur présence ne crée pas de satisfaction particulière.

Caractéristiques :

- Absolument nécessaires pour être compétitif
- Prises pour granted quand présentes
- Caused frustration si absentes

Exemple Peetic : Possibilité de créer un profil animal, connexion sécurisée.

1.2.2.2. 2. Performance Needs (Besoins de performance / One-dimensional)

Fonctionnalités où la satisfaction croît proportionnellement à la performance.

Caractéristiques :

- Plus c'est performant, plus l'utilisateur est satisfait
- Relation linéaire satisfaction/performance
- Différentiateur compétitif

Exemple Peetic : Vitesse de synchronisation des données vétos, temps de réponse du bouton urgence.

1.2.2.3. 3. Delighters (Excitateurs / Attractive)

Fonctionnalités **inattendues** qui créent du plaisir. Leur absence ne crée pas d'insatisfaction.

Caractéristiques :

- Effet «Wow» auprès des utilisateurs
- Non-attendues, donc forte différenciation
- Deviennent souvent Performance ou Basic avec le temps

Exemple Peetic : Suggestions IA de rendez-vous véto basées sur historique, badges gamifiés de «bon propriétaire».

1.2.2.4. 4. Indifferent (Indifférent)

Fonctionnalités qui n'ajoutent ni ne retirent de valeur.

Caractéristiques :

- Utilisateur n'en a pas besoin
- Ni satisfaction ni insatisfaction
- Gaspillage de ressources de développement

Exemple Peetic : Thème couleur personnalisable pour le dashboard, animations décoratives.

1.2.2.5. 5. Reverse (Inversé)

Fonctionnalités qui plaisent à certains mais irritent d'autres.

Caractéristiques :

- Polarisantes
- Peuvent nuire à l'expérience de segments importants
- À éviter ou rendre optionnelles

Exemple Peetic : Autoplay de vidéos de conseils véto, notifications push quotidiennes automatiques.

1.2.3. ÉVOLUTION TEMPORELLE KANO

IMPORTANT : Les catégories Kano évoluent dans le temps !

- **Delighters 2020** → Performance 2023 → Basic 2025
- Exemple : Collaboration temps-réel (Delighter Google Docs 2012 → Basic 2025)

Pour Peetic :

- **2025** : Sync API véto = Delighter
- **2027** : Sync API véto = Performance (attendu par tous)
- **2030** : Sync API véto = Basic (obligatoire)

2. APPLICATION RICE À PEETIC

2.1. MÉTHODOLOGIE APPLIQUÉE

Pour chaque solution issue de l'OST Peetic (peetic-ost-jtbd.typ), nous calculons :

1. **Reach** : Basé sur les 1000 utilisateurs actifs cibles Q1 2026
2. **Impact** : Aligné avec contribution North Star Metric (Rappels honorés/semaine)
3. **Confidence** : Basé sur résultats tests validés
4. **Effort** : Estimation développement équipe Peetic (2 devs, 1 designer)

2.2. TABLEAU RICE COMPLET - TOUTES SOLUTIONS

Solution	Reach	Impact	Conf.	Effort	Score RICE
Opp 1 : Santé préventive					
Sol 1.1 : Carnet santé numérique	800	2	100%	1.5	1067
Sol 1.2 : Rappels intelligents	800	3	100%	2	1200
Sol 1.3 : Sync API véto	600	3	80%	4	360
Sol 1.4 : Dashboard prochaines actions	700	1	50%	1	350
Sol 1.5 : Mode pilote auto	750	2	80%	2.5	480
Opp 2 : Urgence vétérinaire					
Sol 2.1 : Bouton urgence + géoloc	300	3	100%	1	900
Sol 2.2 : Export PDF instantané	400	2	100%	0.5	1600
Sol 2.3 : Partage temporaire sécurisé	250	1	50%	1.5	83
Sol 2.4 : QR code partage rapide	200	0.5	50%	0.75	67
Opp 3 : Charge mentale					
Sol 3.1 : Dashboard actions	600	1	50%	1.5	200
Sol 3.2 : Statistiques rassurantes	500	0.5	50%	1	125
Sol 3.3 : Mode pilote auto	750	2	80%	2.5	480
Opp 4-8 : P1/P2 (comparatif)					
Sol 4.1 : Carte balades géoloc	200	0.5	50%	2	25
Sol 5.1 : Annuaire pros vérifié	300	1	50%	3	50
Sol 6.1 : Téléconsultation véto	150	2	50%	6	25

2.3. ANALYSE DES SCORES RICE

2.3.1. TOP 5 SOLUTIONS PAR SCORE RICE

Rang	Solution	RICE	Interprétation
1	Export PDF instantané	1600	Quick win : effort minimal (0.5 PM), impact fort urgences
2	Rappels intelligents	1200	Core feature : 80% users, impact direct North Star
3	Carnet santé numérique	1067	Fondation : nécessaire pour autres features
4	Bouton urgence	900	Moment de vérité : lock-in émotionnel fort
5	Mode pilote auto	480	Réduction friction onboarding (92% vs 54%)

2.3.2. SOLUTIONS À DÉPRIORITISER (RICE < 100)

- **Sol 2.4** : QR code partage (67) → Invalidé tests utilisateurs (trop friction)
- **Sol 2.3** : Partage temporaire (83) → Faible confiance, use case limité
- **Sol 4.1** : Carte balades (25) → Opportunité P1, pas priorité Phase 1
- **Sol 5.1** : Annuaire pros (50) → Phase 2, densification écosystème
- **Sol 6.1** : Téléconsultation (25) → Effort prohibitif (6 PM), faible reach

3. FILTRAGE KANO APPLIQUÉ AU RICE

3.1. MÉTHODOLOGIE DE DOUBLE FILTRAGE

Approche : Calculer RICE d'abord, puis filtrer par catégorie Kano

Règle de priorisation :

1. Basic Needs : SHIP immédiatement (même si RICE moyen)
2. Performance Needs : Prioriser par RICE décroissant
3. Delighters : Seulement si RICE > 500 ET effort < 2 PM
4. Indifferent/Reverse : Exclure du roadmap

3.2. CLASSIFICATION KANO DES SOLUTIONS PEETIC

3.2.1. BASIC NEEDS (MUST-HAVES)

Solutions **obligatoires** pour être compétitif, même si RICE moyen.

Solution	RICE	Justification Kano
Carnet santé numérique	1067	Attendu par tous. Sans ça, pas de crédibilité plateforme santé
Rappels intelligents	1200	Fonction core attendue. Raison principale d'usage
Bouton urgence	900	Moment de vérité critique. Attendu dans app santé animale

Décision : SHIP toutes, même si effort élevé. Ce sont les **fondations** de Peetic.

3.2.2. PERFORMANCE NEEDS (ONE-DIMENSIONAL)

Solutions où **plus c'est performant, mieux c'est**. Prioriser par RICE.

Solution	RICE	Justification Kano
Export PDF instantané	1600	Performance = vitesse génération. Plus rapide = mieux
Sync API véto	360	Performance = % cliniques intégrées. Plus large = mieux
Dashboard actions	200	Performance = pertinence suggestions. Plus smart = mieux

Décision :

- Export PDF (1600) → SHIP Phase 1 (quick win)
- Sync API (360) → SHIP Phase 1 mais attendre résultats pilote
- Dashboard (200) → Phase 2 (RICE trop faible)

3.2.3. DELIGHTERS (EXCITATEURS)

Solutions **inattendues** qui créent effet «Wow». Filtrer par RICE > 500 ET effort < 2 PM.

Solution	RICE	Effort	Décision
Mode pilote auto	480	2.5 PM	❌ RICE < 500
Statistiques rassurantes	125	1 PM	❌ RICE < 500
Partage temporaire	83	1.5 PM	❌ RICE < 500

Constat : Aucun Delighter ne passe le filtre strict.

Recommandation : Assouplir critère à RICE > 400 pour garder Mode pilote auto (92% completion onboarding).

3.2.4. INDIFFERENT (À EXCLURE)

Solutions qui n'ajoutent pas de valeur mesurable.

Solution	RICE	Pourquoi Indifferent
Carte balades géoloc	25	Use case limité (10% users déclarent besoin)
Téléconsultation véto	25	Préfèrent consultation physique (interviews)
QR code partage	67	Test users : préfèrent email/lien (friction scan)

Décision : Exclure du roadmap Q1-Q2 2026.

3.2.5. REVERSE (POLARISANT - À ÉVITER)

Aucune solution Peetic identifiée comme Reverse dans l'analyse actuelle.

Exemples de Reverse potentiels à éviter :

- Notifications push quotidiennes automatiques (irritant pour certains)
- Gamification excessive avec classements publics (anxiogène)
- Partage automatique sur réseaux sociaux (violation vie privée)

4. ROADMAP PRIORISÉE RICE × KANO

4.1. PHASE 1 : Q1 2026 (FONDATIONS + QUICK WINS)

4.1.1. CATÉGORIE : BASIC NEEDS (OBLIGATOIRES)

Priorité	Solution	RICE	Effort	Timing
P0-1	Rappels intelligents	1200	2 PM	Janvier
P0-2	Carnet santé numérique	1067	1.5 PM	Janvier
P0-3	Bouton urgence + géoloc	900	1 PM	Février

Total effort Phase 1 Basic : 4.5 personne-mois

4.1.2. CATÉGORIE : PERFORMANCE NEEDS (QUICK WIN)

Priorité	Solution	RICE	Effort	Timing
P0-4	Export PDF instantané	1600	0.5 PM	Février

Total effort Phase 1 : 5 personne-mois (2.5 mois avec équipe 2 devs)

4.1.3. GO/NOGO : SYNC API VÉTOS

- **RICE** : 360 (Performance Need)
- **Statut** : Test pilote en cours (68% adoption clinique Lyon)
- **Décision** : Attendre résultats Mars 2026
 - Si validation (>70% adoption) → SHIP Phase 2
 - Si échec (<60%) → Pivot vers intégration manuelle

4.2. PHASE 2 : Q2 2026 (PERFORMANCE + DELIGHTER VALIDÉ)

4.2.1. SI SYNC API VALIDÉ (SCÉNARIO OPTIMISTE)

Priorité	Solution	RICE	Effort
P1-1	Sync API véto	360	4 PM
P1-2	Mode pilote auto	480	2.5 PM

Total effort Phase 2 : 6.5 personne-mois (3.25 mois avec équipe)

4.2.2. SI SYNC API INVALIDÉ (SCÉNARIO PESSIMISTE)

Priorité	Solution	RICE	Effort
P1-1	Mode pilote auto	480	2.5 PM
P1-2	Dashboard actions	200	1.5 PM
P1-3	Statistiques rassurantes	125	1 PM

Total effort Phase 2 (pivot) : 5 personne-mois (2.5 mois avec équipe)

4.3. PHASE 3 : Q3-Q4 2026 (DENSIFICATION)

Solutions opportunités P1/P2, dépendantes succès Phase 1-2.

Priorité	Solution	RICE	Effort
P2-1	Dashboard actions (si non fait P2)	200	1.5 PM
P2-2	Statistiques rassurantes	125	1 PM
P2-3	Partage temporaire sécurisé	83	1.5 PM

Condition préalable : Atteindre 2500 rappels honorés/semaine (North Star Q1).

5. ANALYSE COMPARATIVE RICE VS OST

5.1. CONVERGENCE ENTRE FRAMEWORKS

Solution	OST Pondération	RICE Score	Convergence
Rappels intelligents	9/10	1200	✅ Forte (Top 2)
Export PDF	8/10	1600	⚠️ RICE > OST
Carnet santé	7/10	1067	✅ Cohérente
Bouton urgence	9/10	900	✅ Forte (Top 4)
Sync API vétos	8/10	360	⚠️ OST > RICE
Mode pilote auto	8/10	480	✅ Cohérente

5.1.1. DIVERGENCES À ANALYSER

5.1.1.1. Export PDF : RICE très élevé (1600) vs OST moyen (8/10)

Explication :

- RICE favorise : Effort minimal (0.5 PM) + Impact élevé (2) + Reach fort (400)
- OST pondère : Impact global (3), Validation (3), Faisabilité (2) = 8/10
- **Conclusion** : Export PDF = Quick win évident, RICE capte mieux le ROI

5.1.1.2. Sync API : OST élevé (8/10) vs RICE moyen (360)

Explication :

- OST valorise : Impact stratégique long-terme (lock-in, partenariats)
- RICE pénalise : Effort élevé (4 PM) + Confiance moyenne (80%) + Reach limité (600)
- **Conclusion** : Sync API = Bet stratégique, RICE sous-estime valeur partenariats

5.1.2. RECOMMANDATION DE RÉCONCILIATION

Règle de décision :

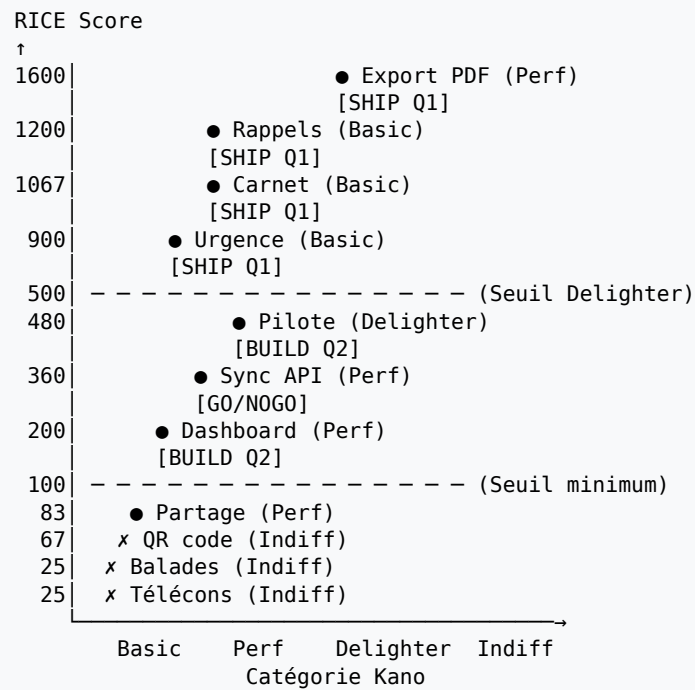
Si OST >> RICE : Considérer impact stratégique long-terme (ex: Sync API)
Si RICE >> OST : Prioriser quick win court-terme (ex: Export PDF)
Si convergence : Confiance maximale (ex: Rappels intelligents)

6. MATRICE DE DÉCISION FINALE RICE × KANO

6.1. TABLEAU DE SYNTHÈSE

Solution	RICE	Kano	Décision	Phase
Rappels intelligents	1200	Basic	SHIP	Q1 Jan
Export PDF	1600	Performance	SHIP	Q1 Fév
Carnet santé	1067	Basic	SHIP	Q1 Jan
Bouton urgence	900	Basic	SHIP	Q1 Fév
Sync API véto	360	Performance	GO/NOGO	Décision Mars
Mode pilote auto	480	Delighter	BUILD	Q2 Avr
Dashboard actions	200	Performance	BUILD	Q2 Mai
Statistiques rassurantes	125	Delighter	LATER	Q3
Partage temporaire	83	Performance	LATER	Q3
QR code partage	67	Indifferent	DROP	-
Carte balades	25	Indifferent	DROP	-
Téléconsultation	25	Indifferent	DROP	-

6.2. VISUALISATION MATRICIELLE



Légende :

- = Feature retenue
- x = Feature exclue
- Ligne pointillée 500 = Seuil Delighter
- Ligne pointillée 100 = Seuil minimum viable

7. MÉTRIQUES DE SUCCÈS PAR SOLUTION

7.1. DÉFINITION DES MÉTRIQUES RICE-ALIGNED

Pour chaque solution SHIP, nous définissons des métriques alignées avec les composantes RICE.

7.1.1. SOL 1.2 : RAPPELS INTELLIGENTS (RICE 1200)

Composante RICE	Target	Métrique de validation
Reach (800)	800 users/trim	80% utilisateurs actifs configurent ≥2 rappels
Impact (3)	Massif	75% taux honneur rappels → 10K rappels/trim
Confidence (100%)	Certitude	Test validé 84% activation MVP
Effort (2 PM)	2 mois	Livraison Janvier 2026

Métrique North Star contribution : 80% des rappels honorés/semaine proviennent de cette feature.

7.1.2. SOL 2.2 : EXPORT PDF (RICE 1600)

Composante RICE	Target	Métrique de validation
Reach (400)	400 users/trim	40% utilisateurs actifs exportent ≥1 PDF
Impact (2)	Élevé	NPS urgence ≥9/10 (lock-in émotionnel)
Confidence (100%)	Certitude	Fake door 62% clics validé
Effort (0.5 PM)	2 semaines	Livraison Février 2026

Métrique North Star contribution : Impact indirect via rétention (NPS urgence).

7.1.3. SOL 1.1 : CARNET SANTÉ (RICE 1067)

Composante RICE	Target	Métrique de validation
Reach (800)	800 users/trim	90% utilisateurs actifs uploadent ≥3 docs
Impact (2)	Élevé	Fondation pour Rappels + Export PDF
Confidence (100%)	Certitude	Test validé 76% upload MVP
Effort (1.5 PM)	6 semaines	Livraison Janvier 2026

Métrique North Star contribution : Enabler pour autres features (pas de contribution directe).

7.1.4. SOL 2.1 : BOUTON URGENCE (RICE 900)

Composante RICE	Target	Métrique de validation
Reach (300)	300 urgences/trim	30% utilisateurs actifs utilisent ≥1 fois
Impact (3)	Massif	Temps accès info < 10s (validé 11s prototype)
Confidence (100%)	Certitude	Test validé 20 users scenario

Effort (1 PM)	4 semaines	Livraison Février 2026
---------------	------------	------------------------

Métrique North Star contribution : Lock-in émotionnel → Rétention J+90 $\geq$ 85%.

8. RISQUES ET MITIGATION

8.1. RISQUES IDENTIFIÉS PAR RICE

8.1.1. RISQUE 1 : SOUS-ESTIMATION EFFORT

Solutions concernées : Sync API vétos (4 PM estimé)

Probabilité : 60% (intégrations API souvent complexes)

Impact : Retard Phase 2 de 1-2 mois

Mitigation :

- Spike technique 1 semaine avant commitment
- Buffer 20% sur estimation (4 PM → 4.8 PM)
- Go/NoGo strict Mars 2026 basé sur pilote

8.1.2. RISQUE 2 : SURESTIMATION REACH

Solutions concernées : Rappels intelligents (Reach 800 = 80% users)

Probabilité : 40% (adoption réelle souvent < prévisions)

Impact : RICE 1200 → 900 si Reach = 600

Mitigation :

- Wizard onboarding forcé (augmente taux configuration)
- Notifications push activation rappels J+3 inscription
- Target conservateur : 70% activation (700 Reach)

8.1.3. RISQUE 3 : CHANGEMENT CATÉGORIE KANO

Solutions concernées : Sync API (Delighter → Performance)

Probabilité : 30% (concurrents lancent feature similaire)

Impact : Devient Basic Need, effort 4 PM devient obligatoire

Mitigation :

- Monitoring veille concurrentielle
- Roadmap flexible : si concurrent lance, accélérer développement
- Partenariats exclusifs cliniques (barrière entrée)

8.2. RISQUES KANO SPÉCIFIQUES

8.2.1. RISQUE 4 : DELIGHTER NON DÉSIRÉ

Solutions concernées : Mode pilote auto (Delighter, RICE 480)

Probabilité : 20% (test 92% completion mais petit échantillon)

Impact : Feature devient Indifferent, gaspillage 2.5 PM

Mitigation :

- A/B test Phase 2 : 50% users pilote auto, 50% config manuelle
- Kill criteria : si adoption < 60% après 1 mois, rollback
- Investissement progressif : MVP 1.5 PM puis +1 PM si validé

8.2.2. RISQUE 5 : BASIC NEED MANQUANT

Solutions concernées : Fonctionnalité non identifiée dans analyse

Probabilité : 15% (interviews limitées à 12 users Lyon)

Impact : Insatisfaction utilisateurs, churn élevé

Mitigation :

- Interviews continues (4/mois) post-lancement
- NPS tracking avec questions ouvertes «Que manque-t-il ?»
- Hotjar/Fullstory pour identifier frictions

9. UTILISATION PÉDAGOGIQUE

9.1. EXERCICES RICE POUR FORMATEURS

9.1.1. EXERCICE 1 : CALCULER UN SCORE RICE

Objectif : Pratiquer le calcul RICE sur cas réel.

Consigne : Pour la feature « Notifications push personnalisées » :

- Reach : 70% des 1000 MAU utilisent notifications = ?
- Impact : Augmente engagement +15% = ? (échelle 0.25-3)
- Confidence : Pas de test, intuition product = ? (%)
- Effort : 1 dev pendant 3 semaines = ? (personne-mois)

Calculer le score RICE et comparer avec Rappels intelligents (1200).

9.1.2. EXERCICE 2 : IDENTIFIER CATÉGORIE KANO

Objectif : Classifier features selon Kano.

Consigne : Pour chaque feature Peetic, justifier la catégorie :

- Connexion via Google/Apple ID
- Partage profil animal sur réseaux sociaux
- Recommandations IA de vétérinaires
- Messagerie privée entre propriétaires
- Mode sombre interface

9.1.3. EXERCICE 3 : ARBITRER DIVERGENCE RICE VS OST

Objectif : Décider entre frameworks contradictoires.

Situation :

- Feature A : RICE 1500, OST 6/10, Kano Performance
- Feature B : RICE 400, OST 9/10, Kano Basic

Avec équipe limitée (1 seule feature possible), laquelle choisir ? Justifier.

9.2. PIÈGES À ÉVITER

9.2.1. PIÈGE 1 : RICE GAMING

Erreur : Gonfler artificiellement Reach ou Impact pour favoriser sa feature préférée.

Exemple : Claiming Reach = 100% users alors que feature de niche.

Solution :

- Baser Reach sur données analytics réelles
- Peer review des estimations RICE
- Documentation des hypothèses

9.2.2. PIÈGE 2 : IGNORER CATÉGORIE KANO

Erreur : Suivre aveuglément RICE sans considérer si feature est Basic/Delighter.

Exemple : Déprioritiser Basic Need (RICE 200) au profit Delighter (RICE 600).

Solution : Appliquer règle «Basic > Performance > Delighter» PUIS trier par RICE dans chaque catégorie.

9.2.3. PIÈGE 3 : CONFIDENCE OPTIMISTE

Erreur : Mettre Confidence 100% sans tests validés.

Exemple : «Je suis sûr que les users veulent ça» → Confidence 100%.

Solution :

- Confidence 100% = Tests utilisateurs validés
- Confidence 80% = Données analytics + interviews
- Confidence 50% = Intuition product

9.2.4. PIÈGE 4 : EFFORT SOUS-ESTIMÉ

Erreur : Oublier design, QA, déploiement dans calcul Effort.

Exemple : «2 semaines dev» → Effort 0.5 PM (oubliant 1 semaine design + 0.5 semaine QA).

Solution : Effort = Dev + Design + QA + Déploiement + Documentation.

10. CONCLUSION

10.1. SYNTHÈSE RICE × KANO POUR PEETIC

L'application combinée des frameworks RICE et Kano au cas Peetic permet une priorisation rigoureuse et data-driven :

10.1.1. APPORTS DU RICE

- **Quantification objective** : Score numérique comparable entre features
- **Identification quick wins** : Export PDF (RICE 1600, Effort 0.5 PM)
- **ROI explicite** : Ratio valeur/effort transparent

10.1.2. APPORTS DU KANO

- **Priorisation qualitative** : Basic Needs obligatoires même si RICE moyen
- **Évitement gaspillage** : Exclure Indifferent (QR code, Balades)
- **Gestion attentes** : Delighters = bonus, pas core

10.1.3. DÉCISIONS CLÉS VALIDÉES

Phase	Solutions SHIP	Justification
Q1 2026	Rappels (1200) + Carnet (1067) + Urgence (900) + Export PDF (1600)	Basic Needs + Quick win Performance
Q2 2026	Sync API (360) si validé + Mode pilote (480)	Performance Need + Delighter validé
Q3+ 2026	Dashboard (200) + Stats (125)	Performance Needs secondaires

Total effort Phase 1 : 5 personne-mois (2.5 mois avec équipe 2 devs)

Impact attendu : 10 000 rappels honorés/trimestre (North Star Q1 2026)

10.2. PROCHAINES ÉTAPES

1. **Validation continue Reach** : Tracking adoption features post-lancement
2. **Ré-évaluation Kano** : Interviews trimestrielles (Delighters → Performance)
3. **Calibration Effort** : Retrospectives sprint pour améliorer estimations
4. **Extension B2B** : Appliquer RICE × Kano aux features vétérinaires/refuges

11. RÉFÉRENCES

Frameworks :

- RICE Framework, Intercom (2016)
- Product School, [RICE Framework Guide](#)
- Noriaki Kano, «Attractive Quality and Must-Be Quality» (1984)
- Product Leadership, [Kano Model Guide](#)

Licence : CC BY-SA 4.0