

Quiz ISEN

Installation : Node compatible Expo SDK 57, puis `npm ci` dans ce dossier. Lancement web : `npm run web`.
Téléphone : `npm start`, puis Expo Go compatible. Tests : `npm test`. Export web : `npm run export`.

Étapes

Commande	Fichier actif	Usage
<code>npm run etape -- 00</code>	<code>etapes/00-depart/App.js</code>	S1 : squelette à compléter
<code>npm run etape -- 01</code>	<code>etapes/01-milieu/App.js</code>	S1 : une question, état du message
<code>npm run etape -- 02</code>	<code>etapes/02-fin-seance1/App.js</code>	S1 : trois questions, score, redémarrage
<code>npm run etape -- 03</code>	<code>etapes/03-composants/App.js</code>	S2 : extraction du bouton Action
<code>npm run etape -- 04</code>	<code>etapes/04-navigation/App.js</code>	S3 : quatre écrans, paramètres
<code>npm run etape -- 05</code>	<code>etapes/05-stockage/App.js</code>	S4 : stockage et gestion des erreurs
<code>npm run etape -- 06</code>	<code>etapes/06-reference/App.js</code>	S5 : référence finale de recette

Les étapes 05 et 06 ont les mêmes fonctionnalités : 06 est la copie de référence stable utilisée pour les tests. Les étapes partagent uniquement les dépendances et le point d'entrée ; leurs fichiers métier sont distincts. Modifier le fichier de l'étape, pas `App.js` à la racine. Le sélecteur refuse d'écraser un `App.js` personnalisé. Un changement d'étape peut demander un rechargement ou un redémarrage Expo.

Architecture finale

`index.js` enregistre App ; App sélectionne une étape. Dans l'étape finale, les écrans restent regroupés dans `App.js` pour faciliter la démonstration. Home, Quiz, Result et History sont des composants séparés ; `HistoryProvider` partage état et actions. `questions.js` contient les données ; `domain.mjs` valide et borne l'historique ; `storage.js` adapte `AsyncStorage`. Une extraction en fichiers d'écrans constitue un exercice complémentaire.

Une réponse est comptée au plus une fois par question. La fin remplace l'écran Quiz ; Rejouer recrée son état. Quitter une partie l'abandonne. Enregistrer est explicite et désactivé tant que le chargement n'a pas réussi. Les résultats sont conservés sous une clé versionnée, sans nom, avec un maximum de 20 entrées. L'effacement demande une confirmation. Une erreur ne déclenche pas une écriture vide de secours.

`AsyncStorage 2.2.0` est utilisé avec son export par défaut ; ne pas recopier sans adaptation les exemples de la version 3. La liste des versions exactes est verrouillée par `package-lock.json`. Ne pas substituer automatiquement des versions « latest ».

Limites pédagogiques

Pas d'authentification, de serveur, de synchronisation multiappareil, ni de sauvegarde d'une partie en cours. L'historique local n'est pas chiffré. Le navigateur est une cible de développement et de secours ; les tests Android restent distincts. Les tests Node couvrent la logique de données, pas à eux seuls l'interface native.