

POUR ATTEINDRE LE SEUIL DE RÉUSSITE, L'ÉTUDIANT.E PROUVERA QU'IL/ELLE EST CAPABLE :

Acquis d'apprentissage	ACQUIS	NON ACQUIS
AA 1 : AA 1 : Créer et exploiter des scripts clients permettant de répondre au cahier des charges <i>Cet acquis évalue la capacité de l'étudiant à développer un jeu Memory interactif en JavaScript et CSS, en respectant les spécifications fonctionnelles et techniques du projet.</i> AA 1.1 : AA 1.1 : Concevoir et implémenter l'interface utilisateur du jeu ☒ Niveau 1: L'interface utilisateur est incomplète, non fonctionnelle, ou ne respecte aucune des exigences visuelles et structurelles du cahier des charges (écrans manquants, cartes non affichées ou non stylisées). ☐ Niveau 2: L'interface utilisateur est partiellement implémentée mais présente des lacunes significatives (un seul écran fonctionnel, problèmes majeurs de disposition des cartes, manque de réactivité des boutons ou des éléments visuels). ☐ Niveau 3: L'interface utilisateur est entièrement implémentée en HTML et CSS Grid, gère correctement les trois états du jeu (accueil, jeu, fin) avec les éléments visuels requis (logo, titres, cartes, boutons) et une disposition fonctionnelle. ☐ Niveau 4: L'interface utilisateur est complète, esthétiquement soignée, responsive et intègre des animations fluides (flip, hover), démontrant une excellente maîtrise du CSS et de l'expérience utilisateur. AA 1.2 : AA 1.2 : Développer la logique fonctionnelle du jeu Memory ☒ Niveau 1: La logique de jeu est absente ou ne permet pas de jouer du tout (mélange non effectué, cartes non cliquables, paires non détectées, jeu non relançable). ☐ Niveau 2: La logique de jeu est implémentée mais contient des bugs majeurs qui empêchent une partie complète (cartes qui ne se retournent pas correctement, paires non validées, délai d'affichage incorrect, jeu bloqué après quelques tours). ☐ Niveau 3: La logique de jeu (mélange, gestion des clics, retournement des cartes, vérification des paires, délai de 1 seconde, réinitialisation) est entièrement fonctionnelle et respecte toutes les règles de base du cahier des charges. ☐ Niveau 4: La logique de jeu est robuste, gère les cas limites (par exemple double-clic rapide) et intègre avec succès une ou plusieurs fonctionnalités supplémentaires (score, modes de difficulté, sauvegarde des scores) améliorant significativement l'expérience de jeu. AA 1.3 : AA 1.3 : Gérer les interactions utilisateur et les événements DOM ☒ Niveau 1: Aucune interaction utilisateur n'est gérée ou les événements clés (clic sur carte, boutons) sont inopérants, rendant le jeu injouable. ☐ Niveau 2: Les interactions de base sont implémentées mais avec des erreurs (clics multiples sur la même carte, clics désynchronisés, boutons ne déclenchant pas l'action attendue de manière fiable). ☐ Niveau 3: Les interactions utilisateur sont correctement gérées via des écouteurs d'événements JavaScript, assurant une réponse adéquate aux clics sur les cartes et les boutons Jouer/Rejouer sans blocage. ☐ Niveau 4: La gestion des événements est optimisée pour une fluidité maximale, avec des mécanismes pour prévenir les interactions indésirables (désactivation des clics pendant l'animation des cartes) et améliorer l'ergonomie générale du jeu.	☐	☒
AA 2 : AA 2 : Structurer, remettre et exploiter le code du projet <i>Cet acquis évalue la capacité de l'étudiant à organiser son code de manière structurée et lisible, à appliquer des principes de programmation, et à remettre un dépôt GitHub exploitable contenant le code évalué.</i> AA 2.1 : AA 2.1 : Structurer le code et appliquer les principes de la programmation ☒ Niveau 1: Le code est désorganisé, non commenté et difficilement lisible, empêchant la compréhension ou la maintenance du projet. ☐ Niveau 2: Le code est peu structuré, contient des répétitions importantes ou des fonctions monolithiques, rendant son évolution complexe. ☐ Niveau 3: Le code est bien organisé en fonctions distinctes et logiques, avec une indentation cohérente et des commentaires pertinents, démontrant une bonne compréhension des principes de modularité. ☐ Niveau 4: Le code est modulaire, extensible et tire parti de la programmation orientée objet (classes ou constructeurs pour les cartes, le plateau ou le jeu) pour une architecture propre et maintenable. AA 2.2 : AA 2.2 : Remettre un dépôt GitHub exploitable contenant le code évalué ☒ Niveau 1: Aucun lien de dépôt GitHub exploitable n'est remis, le lien est inaccessible, le dépôt est vide, ou aucun code correspondant au travail évalué n'y est présent. ☐ Niveau 2: Un dépôt est remis mais le code est incomplet, ne correspond pas clairement au travail évalué, ou aucun commit identifiable ne permet d'attester la remise du code. ☐ Niveau 3: Le dépôt GitHub est accessible et contient le code complet du projet dans au moins un commit identifiable, permettant d'évaluer le travail remis. ☐ Niveau 4: Le dépôt GitHub est accessible, contient le code complet du projet et présente plusieurs commits cohérents ou significatifs, facilitant le suivi du développement. AA 2.3 : AA 2.3 : Débuguer et utiliser la documentation technique ☒ Niveau 1: L'étudiant ne parvient pas à identifier ou corriger les erreurs de programmation, et ne fait aucune tentative d'utilisation de la documentation ou des outils de débogage. ☐ Niveau 2: L'étudiant identifie quelques erreurs simples mais rencontre des difficultés majeures à les résoudre, ou utilise la documentation de manière limitée et inefficace. ☐ Niveau 3: L'étudiant est capable d'identifier et de corriger la plupart des erreurs de programmation courantes de manière autonome, en s'appuyant sur la console du navigateur et la documentation JavaScript. ☐ Niveau 4: L'étudiant débogue efficacement des problèmes complexes avec des outils de développement avancés (points d'arrêt, analyse des variables) et démontre une capacité à rechercher et appliquer des solutions pertinentes issues de la documentation technique.	☐	☒

Tous les AA sont « acquis »
RÉUSSITE
→ Voir degré de maîtrise

Au moins un AA non-acquis (en 1ère session)
AJOURNEMENT
→ Voir justification ajournement

Au moins un AA non-acquis (en 2ème session)
REFUS
→ Voir justification refus

DEGRÉ DE MAÎTRISE (% ≥ 50) - 0= Acquis, 1 = Bien, 2 = TB, 3 = TTB, 4 = Excellent et 5 = Parfait

Critères	Indicateurs	Niveau
N°1 Qualité de l'expérience utilisateur (UI/UX)	✓ 0/5 (0/14.0) : L'interface est non fonctionnelle ou totalement incompréhensible. Aucune attention n'est portée à l'esthétique ou à la réactivité. L'expérience utilisateur est inexistante. 1/5 (2.8/14.0) : L'interface est rudimentaire et peu attrayante. Des problèmes majeurs d'ergonomie ou de réactivité rendent l'utilisation difficile. Aucune animation visuelle n'est présente. 2/5 (5.6/14.0) : L'interface est fonctionnelle mais manque d'esthétisme et de fluidité. La réactivité est absente ou très limitée, rendant l'expérience utilisateur basique et peu agréable. 3/5 (8.4/14.0) : L'interface est claire et fonctionnelle, avec un design acceptable. Quelques éléments visuels sont présents. La réactivité est partiellement gérée. Des animations simples peuvent être présentes mais manquent de fluidité. 4/5 (11.2/14.0) : L'interface est soignée, intuitive et agréable à utiliser. Elle intègre des animations fluides (flip, hover) et est globalement responsive pour différentes tailles d'écran, offrant une bonne expérience utilisateur. 5/5 (14.0/14.0) : L'interface est esthétiquement remarquable, très intuitive et offre une expérience utilisateur exceptionnelle. Les animations sont parfaitement fluides et le design est entièrement adaptatif, avec une attention méticuleuse aux détails.	0/14.0
N°2 Robustesse et stabilité de l'application	✓ 0/5 (0/14.0) : L'application présente de nombreux bugs bloquants ou des comportements imprévisibles qui rendent le jeu injouable ou impossible à terminer, même pour une seule partie. 1/5 (2.8/14.0) : L'application est très instable, avec des bugs fréquents qui perturbent gravement le déroulement du jeu (blocages, comportements inattendus, cartes qui ne se retournent pas ou restent bloquées). 2/5 (5.6/14.0) : L'application fonctionne mais présente des bugs récurrents qui nuisent à l'expérience de jeu (possibilité de retourner plus de deux cartes, problèmes de réinitialisation, comportements incohérents). 3/5 (8.4/14.0) : L'application est généralement stable, mais quelques bugs mineurs ou cas limites peuvent apparaître (clics très rapides, réinitialisation imparfaite après une victoire, latence inattendue). 4/5 (11.2/14.0) : L'application est robuste et stable. Les principaux cas d'utilisation sont bien gérés, y compris les interactions rapides ou la réinitialisation de plusieurs parties, sans bugs majeurs perturbant le jeu. 5/5 (14.0/14.0) : L'application est d'une excellente robustesse, gérant impeccablement les cas limites et scénarios complexes (anti-spam de clics, réinitialisation parfaite après de nombreuses parties). Le jeu est fluide et fiable en toutes circonstances.	0/14.0
N°3 Implémentation des fonctionnalités supplémentaires	✓ 0/5 (0/14.0) : Aucune fonctionnalité supplémentaire n'est implémentée ou les tentatives sont totalement non fonctionnelles et n'apportent aucune valeur ajoutée. 1/5 (2.8/14.0) : Une fonctionnalité supplémentaire est tentée mais n'est pas fonctionnelle, est implémentée de manière très incomplète, ou présente des bugs majeurs. 2/5 (5.6/14.0) : Une fonctionnalité supplémentaire est implémentée et partiellement fonctionnelle, ou plusieurs sont tentées avec des lacunes importantes et une intégration peu soignée. 3/5 (8.4/14.0) : Deux fonctionnalités supplémentaires sont implémentées et fonctionnelles, même si elles peuvent présenter quelques imperfections mineures dans leur intégration ou leur robustesse. 4/5 (11.2/14.0) : Trois fonctionnalités supplémentaires sont implémentées, fonctionnelles et bien intégrées au jeu, améliorant significativement l'expérience utilisateur et démontrant une bonne maîtrise technique au-delà des exigences de base. 5/5 (14.0/14.0) : Quatre fonctionnalités supplémentaires ou plus sont implémentées avec succès, de manière robuste et élégante, démontrant une maîtrise avancée et une réelle valeur ajoutée au jeu (design soigné complet, score détaillé, plusieurs modes de difficulté, sauvegarde localStorage).	0/14.0
N°4 Qualité et maintenabilité du code	✓ 0/5 (0/8) : Le code est totalement illisible, sans structure logique, sans indentation et sans commentaires, rendant toute compréhension ou modification impossible. 1/5 (1.6/8) : Le code est très désordonné, contient des répétitions excessives et des fonctions monolithiques. Les commentaires sont absents ou inutiles. Les conventions de base sont ignorées. 2/5 (3.2/8) : Le code est partiellement structuré, mais certaines sections sont difficiles à suivre. Il y a des redondances et le respect des conventions de nommage ou d'indentation est incohérent. Les commentaires sont rares.	0/8

Critères	Indicateurs	Niveau
	3/5 (4.8/8) : Le code est globalement propre, avec une structure fonctionnelle claire. L'indentation est cohérente et des commentaires pertinents sont présents pour les parties complexes. La modularité est acceptable, mais des améliorations sont possibles. 4/5 (6.4/8) : Le code est propre, bien organisé, modulaire et facile à comprendre. Il respecte la plupart des bonnes pratiques de codage (nommage, fonctions claires, DRY) et est relativement facile à maintenir ou à faire évoluer. 5/5 (8/8) : Le code est exemplaire : très modulaire, élégant, optimisé et extensible. Il respecte rigoureusement les conventions de codage, est abondamment et pertinemment commenté, et démontre une excellente conception logicielle.	

CALCUL DE LA NOTE FINALE		
Seuil de réussite (tous AA acquis) :		0 / 50
Total des critères de maîtrise :		0 / 50
NOTE FINALE :		0 / 100

Date de délibération : 26/06/2026	Signature du/des chargé(s) de cours :
-----------------------------------	---------------------------------------