



Plan de test

MonkeyIsland - DDMP

VERSION : 1.0

RÉVISION : 274

Équipe DDMP

Responsable du document : Clément Pabst

Sommaire

Liste des tableaux

Chapitre 1

Introduction

1.1 Objet

Ce document décrit l'activité de test qui sera menée par MonkeyIsland_DDMP durant le « COOL QT » dans le but de valider le produit « Monkey Island » (MoI). Il est rédigé sous la responsabilité de Clément Pabst et sous la direction du Chef de Projet : Guillaume Muret.

1.2 Portée

Sont concernés par ce document :

- **les testeurs** : afin que ceux-ci sachent ce qu'ils vont tester, comment ils le testent et comment ils rendent compte des résultats de ces tests ;
- **les développeurs** : à titre informatif, afin que ceux-ci sachent comment va être validée leur production ; à titre indicatif afin qu'ils sachent, par la description de la gestion des anomalies, comment ils s'interfaceront avec l'équipe de test ;
- **les professeurs** : ce plan de test, ainsi que son implication, feront l'objet d'audits par le corps professoral.

1.3 Copyright

Le présent document est la propriété de MonkeyIsland_DDMP. Il est diffusé pour les seuls besoins du projet concerné. Il ne doit pas être reproduit, entièrement ou partiellement, ou employé pour tout autre but sans autorisation écrite préalable de Clément Pabst ou Guillaume Muret, et à la condition que cette notification soit incluse dans une telle reproduction. Aucune information quant au contenu ou aux thèmes de ce document ne peut être communiquée de quelque façon à un tiers sans autorisation écrite de Clément Pabst ou Guillaume Muret. L'équipe professorale de l'ESEO en charge du projet Monkey Island 2017/2018 échappe à ces restrictions et peut utiliser le présent document pour toute utilisation qui leur convient dans le contexte du projet Monkey Island, et ce sans autorisation.

1.4 Termes et abréviations

Voici les termes et abréviations nécessaires à la compréhension de l'activité de test :

Terme ou abréviation	Signification
TU (Test unitaire)	Test sur une partie précise du programme visant à contrôler la fiabilité des unités logicielles développées, vérifier le respect de leurs conceptions et à identifier les erreurs logiques.
TI (Test d'intégration)	Test sur un ensemble de fonctions visant à démontrer la stabilité et la cohérence des interfaces et des interactions des unités logicielles et à vérifier le respect de la conception générale.
TV (Test de validation)	Test visant à valider l'adéquation aux spécifications logicielles
DT (Données de test)	Entrées du programme à tester
TC (Cas de test)	Regroupe deux séquences d'action : une pour une pré-condition et une autre pour utiliser les données de test d'un jeu de test
JT (Jeux de test)	Ensemble des données de test
Valider	S'assurer que le logiciel ait les fonctions attendues
Vérifier	S'assurer que le logiciel fonctionne correctement
Test nominal	Test avec des données d'entrée valides
Test de robustesse	Test avec des données d'entrée invalides
L'école	L'ESEO

TABLE 1.1 – Table des termes et abréviations

Chapitre 2

Références

2.1 Normes et standards

Norme	Lien avec le plan de test	Version	Source
Norme 829	Le plan de test s'inspire de ce standard	2008	IEEE

TABLE 2.1 – Table des normes et standards

Chapitre 3

Périmètre de test

3.1 Composants

3.1.1 Composants concernés par les tests

Seront concernés par l'activité de test les composants logiciels développés durant le projet :

- **Serveur** : le logiciel serveur.

3.1.2 Composants non concernés par les tests

Les supports d'exécutions matériels ne seront pas concernés par les tests.

Les supports d'exécutions logiciels suivant ne seront pas concernés par les tests :

- **Le client**, en effet, seul le serveur est testé.

Les supports de communication matériels suivant ne seront pas concernés par les tests :

- **Le réseau de l'école.**

Les supports de communication logiciels :

- **la pile TCP/IP.**

3.2 Fonctionnalités

3.2.1 Incrément 1

Fonctionnalités à tester	Testé	Validation
Déplacement des singes erratiques	oui	A
Gestion des pirates (déplacement et décès)	oui	A

TABLE 3.1 – Table des fonctionnalités testées pour l'incrément 1

Déplacement des singes erratiques	Testé	Validation
Déplacement ok terre vide H-B-G-D	oui	A
Pas de déplacement en diagonale terre vide	oui	A
Obligation de déplacement terre vide	oui	A
Pas de déplacement possible (mer)	oui	A
Pas de déplacement possible (erratiques)	oui	A
Pas de déplacement possible (chasseurs)	oui	A
Déplacement ok - trésor caché -> caché	oui	A
Déplacement ok - pirate vivant -> mort	oui	A
Déplacement ok - pirate mort -> mort	oui	A
Déplacement ok - rhum dispo -> dispo	oui	A
Equiprobabilité pour 4	oui	A
Equiprobabilité pour 3	oui	A
Equiprobabilité pour 2	oui	A
Non équiprobabilité stricte pour 4	oui	A
Non équiprobabilité stricte pour 3	oui	A
Non équiprobabilité stricte pour 2	oui	A
Schéma non répétitif pour 4	oui	B+
Schéma non répétitif pour 3	oui	B+
Schéma non répétitif pour 2	oui	B+
Pas de déplacement "nuageux"	oui	B+

TABLE 3.2 – Table des tests liés au déplacement des singes erratiques

Gestion des pirates (déplacement et décès)	Testé	Validation
Déplacement ok terre vide H-B-G-D	oui	A
Pas de déplacement en diagonale terre vide	oui	A
Pas de déplacement si instruction = 0-0	oui	A
Pas de déplacement possible (mer)	oui	A
Pas de déplacement possible (pirate)	oui	A
Pas de déplacement possible (vide)	oui	A
Pas de déplacement hors carte	oui	A
Déplacement sur singe erratique -> mort	oui	A
Déplacement sur singe chasseur -> mort	oui	A
Pas de déplacement si mort, case terre	oui	A
Pas de déplacement si mort, case terre et rhum	oui	A
Pas de résurrection d'un pirate mort lors de réapparition de rhum	oui	A
Déplacement sur trésor -> découverte trésor et fin de partie	oui	A
Déplacement sur trésor et singe -> découverte trésor, fin de partie et mort	oui	A

TABLE 3.3 – Table des tests liés à la gestion des pirates (déplacement et décès)

3.2.2 Incrément 2

Fonctionnalités à tester	Testé	Validation
Communication client-serveur	oui	A
Gestion de l'énergie de chaque pirate	oui	A

TABLE 3.4 – Table des fonctionnalités testées pour l'incrément 2

Communication client-serveur	Testé	Validation
Inscription du pirate	oui	A
Déplacements HBGD (4tests)	oui	A
NON-Déplacement de 2 cases HBGD (4tests)	oui	A
NON-Déplacement en diagonale (4tests)	oui	A
NON-Déplacement de 2 cases en diagonale (4tests)	oui	A

TABLE 3.5 – Table des tests liés à la communication client-serveur

Gestion de l'énergie de chaque pirate	Testé	Validation
Mort sans énergie	oui	A
Déplacement sur rhum -> énergie<ENERGIE_MAX	oui	A
Déplacement sur rhum -> énergie>ENERGIE_MAX	oui	A
Déplacement sur rhum -> énergie=ENERGIE_MAX	oui	A

TABLE 3.6 – Table des tests liés à la gestion de l'énergie de chaque pirate

3.2.3 Incrément 3

Fonctionnalités à tester	Testé	Validation
Gestion des parties	oui	A
Gestion des singes chasseurs	oui	A
Gestion de l'alcoolémie du pirate	oui	A

TABLE 3.7 – Table des fonctionnalités testées pour l'incrément 3

Gestion des parties	Testé	Validation
Partie finie -> redémarre si le pirate va sur le trésor	oui	A
Partie finie si plus de pirate sur la carte	oui(visuel)	A
Partie finie -> redémarre si plus de pirate vivant sur la carte	oui(visuel)	A

TABLE 3.8 – Table des tests liés à la gestion des parties

Gestion des singes chasseurs	Testé	Validation
Déplacement ok terre vide H-B-G-D	oui	A
Pas de déplacement en diagonale terre vide	oui	A
Obligation de déplacement terre vide	oui	A
Pas de déplacement possible (mer)	oui	A
Pas de déplacement possible (erratiques)	oui	A
Pas de déplacement possible (chasseurs)	oui	A
Déplacement ok - trésor caché -> caché	oui	A
Déplacement ok - rhum dispo -> dispo	oui	A
Déplacement suis le pirate	oui	A
Déplacement ok - pirate vivant -> mort	oui	A
Déplacement ok - pirate mort -> mort	oui	A

TABLE 3.9 – Table des tests liés à la gestion des singes chasseurs

Gestion de l'alcoolémie du pirate	Testé	Validation
Sobre sur rhum (visible) -> énergie>MAX -> en ébriété	oui	A
Sobre sur rhum (invisible) -> énergieNoChange -> PAS en ébriété	oui	A
En ébriété sur rhum (visible) -> en ébriété	oui	A
En ébriété sur rhum (invisible) -> énergie>MAX -> en ébriété	oui	A
En ébriété sur rhum (invisible) -> énergie=MAX -> PAS en ébriété	oui	A
Pirate sur rhum invisible -> visible -> énergie++	oui	A
En ébriété : Probabilité < 1 d'aller là où le joueur demande	oui	B+

TABLE 3.10 – Table des tests liés à la gestion de l'alcoolémie du pirate

3.3 Critères d'acceptation des tests

Pour juger de la performance d'un test nous utiliserons le système de notation suivant :

Note	Résultats pour des tests nominaux
A	Entre 80% et 100% de réussite
B	Entre 50% et 80%(exclus) de réussite
C	Moins de 50% de réussite

TABLE 3.11 – Table des critères d'acceptation des tests Nominaux

Chapitre 4

Processus et stratégie de test

4.1 Activités

Tests d'Intégration : ils sont composés de tests nominaux entre Serveur et Client(s).

Tests Unitaires : ils sont composés de tests nominaux sur certains modules du composant Serveur.

4.2 Techniques de test

L'activité de test sera constituée d'une analyse partitionnelle pour étudier les cas nominaux. Toutes ces analyses seront regroupées dans le cahier de test (ENT/test/cahier_test).

4.3 Outils de test

Outils	Version	Fonction
JMeter	1.6.1	Logiciel de test utilisé pour tester nos communications
JUnit	4	Framework de test unitaire pour Java
FindBugs-IDEA	1.0.1	Framework de test Java pour IntelliJ IDEA
PowerMock	1.7.3	Framework de test Java utilisé pour bouchonner certaines instances

TABLE 4.1 – Table des outils de test

Chapitre 5

Infrastructure de test

Nom	Type	Version
Réseau informatique	WifiEtudiant	-
Matériel personnel	Ordinateur sous Windows	>= Windows 10
	Ordinateur sous Linux	>= Ubuntu 16.04
Logiciel codage Java	IntelliJ IDEA	>= 2017.2.5

TABLE 5.1 – Table de l’infrastructure de test

Chapitre 6

Documents de test et livrables

Nom	SVN	Format	Livable
Plan de test	<i>SVN/test/plan_test</i>	PDF	oui
Cahier de test	<i>SVN/test/cahier_test</i>	PDF	oui
Code de test	<i>SVN/code</i>	.java	oui

TABLE 6.1 – Table des documents de test et livrables

Chapitre 7

Responsabilités

Le Chef de Projet, Guillaume Muret est responsable des moyens mis à disposition pour mener à bien l'activité de test.

Les Testeurs (cf. Equipe de test, chapitre ??) sont responsables des résultats de test reportés dans le cahier de test.

L'équipe DDMP ne peut pas être tenue pour responsable des répercussions d'une défaillance liée à une fonctionnalité non validée.

Chapitre 8

Équipe de test

Voici les membres de l'équipe qui vont prendre part à l'activité de test :

Prénom	Nom	Rôle
Cailyn	Davies	Testeur
Clément	Pabst	Superviseur de l'activité de test, Testeur
François	de Broch d'Hotelans	Testeur
Guillaume	Muret	Chef de projet, Testeur

TABLE 8.1 – Table de l'équipe de test