

COURS: INF 2050 groupe 20

TITRE: Plan de tests

COMMENTAIRE: TP3

Date de remise: 09/05/21

Auteurs équipe 11: Bogdan Sonnenwirth

SONB01029707

## 1. Exigences Fonctionnalités à tester

Identifiants des fonctionnalités	Description des fonctionnalités
EF – 001	Lire le fichier d'entrée et analyser les informations
EF – 002	Calculer les statistiques
EF – 003	Calculer déductions individuelles et le total (en prenant compte des maximums)
EF – 004	Affichage intelligent des erreurs (explication des contraintes)
EF – 005	Mode prédiction, réinitialisation des statistiques, création/affichage des statistiques et arguments
EF – 006	Créer fichier de sortie

## 2. Suites et cas de tests

Identifiants des fonctionnalités	Identifiants des suites de tests	Description des suites de tests	Nombre de cas de tests
EF – 001	ST – 001	Tester la soumission d'un fichier d'entrée	6
EF – 002	ST – 002	Tester le calcul des statistiques	4
EF – 003	ST – 003	Tester le calcul des déductions individuelles et le total	4
EF – 004	ST – 004	Tester l'affichage des erreurs	3
EF – 005	ST – 005	Tester mode prédiction, réinitialisation des statistiques et création/affichage des statistiques	4
EF – 006	ST – 006	Tester la création de fichier de sortie	1

### 3. Cas de tests

Identifiants des suites de tests	Identifiants des cas de tests	Description des cas de tests	Préconditions	Sortie attendue	Priorité
ST – 001	CT-001	Créer une soumission valide	<ul style="list-style-type: none"> <li>-Dossier comportant 7 caractères. 1<sup>er</sup> caractère est une lettre majuscule entre A et E. Les autres sont des chiffres.</li> <li>-Mois valide. Doit comporter la date en format AAAA-MM avec des valeurs normales</li> <li>-Au moins 1 réclamation. Doit contenir un soin valide (0, 100,150, 175, 200, 300-399, 400, 500, 600, 700)</li> <li>-Date valide. Doit comporter la date en format AAAA-MM-JJ avec des valeurs normales.</li> <li>-La valeur de la variable “mois” doit être égale à la partie AAAA-MM de la variable “date”</li> <li>-Le montant doit être supérieur à 0 et doit contenir un “\$” à la fin</li> </ul>	*Aucun message* mais il y a la création du fichier sortie.json si le code au complet est compilé	Haute
	CT-002	Créer une soumission invalide	Dans le fichier entree.json : -“dossier” : “G111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 100, “date” : “2020-01-01”, “montant” : “100.00\$” }]	Dans le fichier sortie.json : -“message”: “Le champ dossier est invalide”	Haute
	CT-003	Créer une soumission invalide	Dans le fichier entree.json : -“dossier” : “A111111”, -“mois” : “2020-13”, -“reclamations” : [{ “soin” : 100, “date” : “2021-01-01”, “montant” : “100.00\$” }]	Dans le fichier sortie.json : -“message”: “Le champ mois est invalide”	Haute
	CT-004	Créer une soumission invalide	Dans le fichier entree.json : -“dossier” : “A111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 900, “date” : “2020-01-01”, “montant” : “100.00\$” }]	Dans le fichier sortie.json : -“message”: “Le champ soin est invalide”	Haute
	CT-005	Créer une soumission invalide	Dans le fichier entree.json : -“dossier” : “A111111”, -“mois” : “2021-01”, -“reclamations” : [{ “soin” : 100, “date” : “2020-01-01”, “montant” : “100.00\$” }]	Dans le fichier sortie.json : -“message”: “Le champ date associe au 1eme reclamation ne correspond pas au champ mois de la demande”	Haute

	CT-006	Créer une soumission invalide	Dans le fichier entree.json : -“dossier” : “A111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 100, “date” : “2020-01-01”, “montant” : “100.00” }]	Dans le fichier sortie.json : -“message”: “Le champ montant de la 1eme reclamation est invalide”	Haute
ST – 002	CT-007	Calculer les statistiques en ayant un fichier d’entrée valide	Dans le fichier entree.json : -“dossier” : “A111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 0, “date” : “2020-01-01”, “montant” : “130.00\$” }],{ “soin” : 100, “date” : “2020-01-02”, “montant” : “50.00\$” }]	Dans le fichier statistiques.txt : -Le nombre de réclamations valides traitees est 2 -Le nombre de demandes rejetees est 0 -soin 0 : 1 Montant maximal reclame : 130.00\$ Moyenne des montants reclames : 130.00\$ -soin 100 : 1 Montant maximal reclame : 50.00\$ Moyenne des montants reclames : 50.00\$	Moyenne
	CT-008	Calculer les statistiques après avoir compilé un fichier d’entrée valide et un fichier invalide	-Un fichier d’entrée valide (CT-007) -Un fichier d’entrée invalide :  -“dossier” : “ ”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 0, “date” : “2020-01-01”, “montant” : “130.00\$” }],{ “soin” : 100, “date” : “2020-01-02”, “montant” : “50.00\$” }]	Dans le fichier statistiques.txt : -Le nombre de réclamations valides traitees est 2 -Le nombre de demandes rejetees est 1	Moyenne
	CT-009	Calculer les statistiques après avoir compilé un fichier d’entrée valide, un fichier invalide et un fichier d’entrée valide	-Un fichier d’entrée valide (CT-007) -Un fichier d’entrée invalide (CT-008) -Un fichier d’entrée valide (CT-007)	Dans le fichier statistiques.txt : -Le nombre de réclamations valides traitees est 4 -Le nombre de demandes rejetees est 1 -soin 0 : 1 Montant maximal reclame : 130.00\$ Moyenne des montants reclames : 130.00\$ -soin 100 : 1 Montant maximal reclame : 50.00\$ Moyenne des montants reclames : 50.00\$	Moyenne

	CT-010	Calculer les statistiques après avoir compilé un fichier d'entrée valide, un fichier invalide, un fichier d'entrée valide et un fichier invalide	-Un fichier d'entrée valide (CT-007) -Un fichier d'entrée invalide (CT-008) -Un fichier d'entrée valide (CT-007) -Un fichier d'entrée invalide (CT-008)	Dans le fichier statistiques.txt : -Le nombre de réclamations valides traitées est 4 -Le nombre de demandes rejetées est 2	Moyenne
ST – 003	CT-011	Tester le calcul des déductions individuelles et le total en ayant un fichier valide	Dans le fichier entree.json : -"dossier" : "A111111", -"mois" : "2020-01", -"reclamations" : [{ "soin" : 0, "date" : "2020-01-01", "montant" : "130.00\$" }], -"soin" : 100, -"date" : "2020-01-02", -"montant" : "50.00\$" }]	Dans le fichier sortie.json : -"dossier" : "A111111", -"mois" : "2020-01", -"remboursements" : [{ "soin" : 0, "date" : "2020-01-01", "montant" : "32.50\$" }], -"soin" : 100, -"date" : "2020-01-02", -"montant" : "17.50\$" }], -"total" : "50.00\$"	Haute
	CT-012	Tester le calcul des déductions individuelles et le total en ayant un fichier valide	Dans le fichier entree.json : -"dossier" : "D111111", -"mois" : "2020-01", -"reclamations" : [{ "soin" : 377, "date" : "2020-01-01", "montant" : "130.00\$" }], -"soin" : 301, -"date" : "2020-01-02", -"montant" : "50.00\$" }]	Dans le fichier sortie.json : -"dossier" : "D111111", -"mois" : "2020-01", -"remboursements" : [{ "soin" : 377, "date" : "2020-01-01", "montant" : "130.00\$" }], -"soin" : 301, -"date" : "2020-01-02", -"montant" : "50.00\$" }], -"total" : "180.00\$"	Haute
	CT-013	Tester le calcul des déductions individuelles et le total en ayant un fichier valide	Dans le fichier entree.json : -"dossier" : "B111111", -"mois" : "2020-01", -"reclamations" : [{ "soin" : 200, "date" : "2020-01-01", "montant" : "100.00\$" }], -"soin" : 400, -"date" : "2020-01-02", -"montant" : "150.00\$" }]	Dans le fichier sortie.json : -"dossier" : "B111111", -"mois" : "2020-01", -"remboursements" : [{ "soin" : 200, "date" : "2020-01-01", "montant" : "100.00\$" }], -"soin" : 400, -"date" : "2020-01-02", -"montant" : "0.00\$" }], -"total" : "100.00\$"	Haute

	CT-014	Tester le calcul des déductions individuelles et le total en ayant un fichier valide	Dans le fichier entree.json : -“dossier” : “C111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 700, “date” : “2020-01-01”, “montant” : “270.00\$” }, { “soin” : 100, “date” : “2020-01-02”, “montant” : “180.00\$” }]	Dans le fichier sortie.json : -“dossier” : “C111111”, -“mois” : “2020-01”, -“remboursements” : [{ “soin” : 700, “date” : “2020-01-01”, “montant” : “243.00\$” }, { “soin” : 100, “date” : “2020-01-02”, “montant” : “171.00\$” }], -“total” : “414.00\$”	Haute
ST – 004	CT-015	Tester l’affichage des erreurs en ayant un montant réclamé supérieur à 500\$	Dans le fichier entree.json : -“dossier” : “C111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 700, “date” : “2020-01-01”, “montant” : “770.00\$” }]	Dans le fichier sortie.json : -“message” : “Le montant mensuel reclame pour la categorie de soin medecin generaliste prive (175) est superieur de 500.00\$”	Moyenne
	CT-016	Tester l’affichage des erreurs en ayant un montant réclamé inférieur à 0\$	Dans le fichier entree.json : -“dossier” : “C111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 175, “date” : “2020-01-01”, “montant” : “-770.00\$” }]	Dans le fichier sortie.json : -“message” : “Le champ montant de la 1eme reclamation est invalide, car le montant est inferieur a 0.00\$”	Moyenne
	CT-017	Tester l’affichage des erreurs en ayant plus de 4 soins réclamé pour 1 jour	Dans le fichier entree.json : -“dossier” : “C111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 175, “date” : “2020-01-01”, “montant” : “70.00\$” }, { “soin” : 100, “date” : “2020-01-01”, “montant” : “180.00\$” }, { “soin” : 200, “date” : “2020-01-01”, “montant” : “185.00\$” }, { “soin” : 700, “date” : “2020-01-01”, “montant” : “100.00\$” }, { “soin” : 350, “date” : “2020-01-01”, “montant” : “140.00\$” }]	Dans le fichier sortie.json : -“message” : “Erreur! Il y a plus de 4 soins pour la date de 2020-01-01”	Moyenne

ST – 005	CT-018	Tester mode prédiction en ayant -p comme premier argument à la première compilation	Dans le fichier entrée.json : -“dossier” : “A111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 0, “date” : “2020-01-01”, “montant” : “130.00\$” }, { “soin” : 100, “date” : “2020-01-02”, “montant” : “50.00\$” }]	-Le fichier sortie.json ne sera pas considéré pour les stats.  - Le fichier statistiques.txt n'existe pas (seulement à la première compilation).	Moyenne
	CT-019	Tester la réinitialisation des statistiques en ayant -SR comme premier argument	Dans le fichier statistiques.txt : -Le nombre de réclamations valides traitées est 2 -Le nombre de demandes rejetées est 1	Dans le fichier statistiques.txt : -Le nombre de réclamations valides traitées est 0 -Le nombre de demandes rejetées est 0  Dans la console après la compilation : -“Les statistiques sont réinitialisées!”	Moyenne
	CT-020	Tester l’affichage des statistiques en ayant -S comme premier argument	Dans le fichier entrée.json : -“dossier” : “A111111”, -“mois” : “2020-01”, -“reclamations” : [{ “soin” : 0, “date” : “2020-01-01”, “montant” : “130.00\$” }]	Dans la console après la compilation : -Le nombre de réclamations valides traitées est 1 -Le nombre de demandes rejetées est 0 - -soin 0 : 1 Montant maximal réclame : 130.00\$ Moyenne des montants réclames : 270.00\$	Moyenne
	CT-021	Tester du code avec un argument invalide	-Un fichier entree.json valide Arguments : -e entree.json sortie.json	Dans la console après la compilation : -“Le programme n’accepte que l’argument ‘-p’ comme le premier argument, ‘entrée.json’ comme le deuxième argument et ‘sortie.json’ pour le troisième argument lorsqu’il y a 3 arguments. Veuillez recommencer”	Moyenne
ST – 006	CT-022	Créer un fichier de sortie	Avoir un fichier d’entrée	*Aucun message* -Création/changement du fichier sortie.json	Haute