

| |
|--|
| TC7 |
| Tests informatiques |
| TP – Mise en place d’un outil de gestion des tests de validation fonctionnelle |

La réalisation de ce TP sera évaluée

Table des matières

| | | |
|-------|--|----|
| I. | Contexte | 2 |
| III. | Prérequis | 4 |
| IV. | Installation de ProjeQtOr | 4 |
| VI. | Configuration de base | 7 |
| | Paramétrage du projet | 7 |
| | Paramétrage des types de tickets | 8 |
| | Paramétrage des niveaux d’urgence des tickets..... | 8 |
| VII. | Organiser des projets de test dans ProjeQtOr | 9 |
| | Importer les plans de test dans ProjeQtOr | 9 |
| | Définir le produit à tester | 10 |
| | Définir les composants du produit | 10 |
| | Définir la plateforme de test | 10 |
| | Gérer les terminaux de test : les contextes..... | 10 |
| | Le référentiel documentaire | 10 |
| VIII. | Définir un projet de tests | 12 |
| | Introduction et prérequis | 12 |
| | Utiliser la gestion de projet | 12 |
| | Les activités et les jalons | 14 |
| IX. | Organiser les tests dans ProjeQtOr | 16 |
| | Définir les cas de test | 16 |
| | Préparer les sessions des cas de test | 16 |
| | Créer une session : regrouper des cas de test | 16 |
| | Définir l’enchaînement des sessions, les regrouper | 17 |
| X. | Exécuter des tests avec ProjeQtOr | 17 |
| XI. | Organiser la résolution des tickets | 17 |

I. Contexte



Vous travaillez pour les studios WonderWare, une entreprise spécialisée dans la création de jeux vidéo située à Lyon en France. Votre studio développe actuellement un nouveau jeu d'aventure appelé **Secret Things** pour le compte de la multinationale Sunny Entertainment. Ce jeu est appelé à révolutionner l'univers du jeu vidéo grâce à la qualité de ses scénarios, à son graphisme époustouflant et à son univers hyper immersif.

Après un an de travail, la version 1.0 du jeu arrive en phase de recette et fait l'objet de nombreux tests. En complément du jeu, le client a également demandé à WonderWare de développer une nouvelle **plateforme web** destinée à accueillir les futurs joueurs. Ceux-ci pourront s'y connecter pour accéder aux jeux édités par Sunny Entertainment, avoir accès à des bonus, acheter des extensions, discuter avec les autres joueurs, être informés des nouveautés des jeux, etc.



Vous travaillez pour la maîtrise d'ouvrage (MOA) et êtes nommé **chef de projet** pour **les tests de validation fonctionnelle de la plateforme web** auxquels participent la maîtrise d'ouvrage interne (deux personnes) ainsi que trois représentants de Sunny Entertainment venus spécialement de Los Angeles.

Une de vos premières tâches est de mettre en place un outil de suivi des tests fonctionnels. Votre responsable vous oriente vers l'outil ProjeQtOr car celui-ci est utilisé chez WonderWare depuis cinq ans pour gérer les tests fonctionnels.

Votre mission est de paramétrer ProjeQtOr afin qu'il puisse servir de base pour :

- **Organiser les tests fonctionnels entre les participants,**
- **Servir de base pour la gestion des scénarios de test,**
- **Suivre l'exécution des tests,**
- **Recueillir les tickets générés par les testeurs,**
- **Suivre la résolution des tickets.**

Pour ce faire, vous devez suivre les instructions données dans ce TP, mais aussi faire preuve de logique et de sagacité. **C'est principalement votre capacité à comprendre le besoin et à mettre en place un environnement de tests opérationnel qui seront notés.**



II. Prérequis

Vous devez disposer d'une instance de Wampserver opérationnelle.

III. Installation de ProjQtOr

1. Créez un répertoire **projqtor** dans le dossier web de wampserver (normalement **c:\wamp64\www**).
2. Créez un **vhost** pointant sur ce dossier (par exemple **www.testsfonctionnels.com**).
3. Téléchargez l'application **ProjQtOr** à partir du site de son éditeur :

<http://www.projqtor.org/fr/product-fr/downloads-fr>

4. Décompressez le fichier zip puis copier le contenu dans le dossier **projqtor**.
5. Avant de configurer **ProjQtOr**, vérifiez que votre plateforme Wampserver correspond bien à vos attentes, notamment la version MySQL et de PHP.

Les prérequis techniques sont donnés à l'adresse suivante :

<https://www.projqtor.org/files/readme.txt>

6. Tapez l'URL de votre vhost dans votre navigateur. ProjQtOr est alors lancé en mode configuration, ce qui se manifeste par l'apparition du formulaire suivant :

Attention : ce formulaire ne sera affiché qu'une seule fois. Si votre installation est en échec, vous serez peut-être alors contraint d'utiliser phpMyAdmin pour supprimer la base de données (si elle a été créée) dans l'éventualité d'un retour en arrière.

Normalement, l'intégralité des paramètres de configuration sont correctement renseignés par défaut. Vous ne devriez pas avoir à changer quoi que ce soit.

7. Validez ces paramètres de configuration.

Une fois la configuration validée, ProjeQtOr vérifie plusieurs éléments de celle-ci. Le résultat de ce diagnostic est affiché comme suit :



Dans la capture d'écran ci-dessus, les messages en rouge indiquent quelques problèmes : l'extension IMAP n'est pas installée et le paramètre max_input_vars est dimensionné à 1000 caractères, ce qui est estimé insuffisant.

8. Rectifiez au besoin les fichiers de configuration comme indiqué (notamment le fichier php.ini), mais normalement tout devrait bien se passer.

En cas de problème cependant, le formulaire de configuration de ProjeQtOr est précis. Vous saurez donc quoi faire.

Les extensions sont accessibles par un clic sur l'icône Wamp, menu **Php / Extension PHP**.

Retenez que les messages en rouge ne sont pas nécessairement des alertes bloquantes. La validation doit s'achever par "**Database XXX created : OK**" puis "**Parameters are saved.**" où XXX est le nom que vous avez donné à la base.

Si des messages apparaissent en rouge, alors vous devez recommencer votre installation : il est fortement probable que ProjeQtOr n'ait pas réussi à se connecter au serveur de bases de données MySQL.

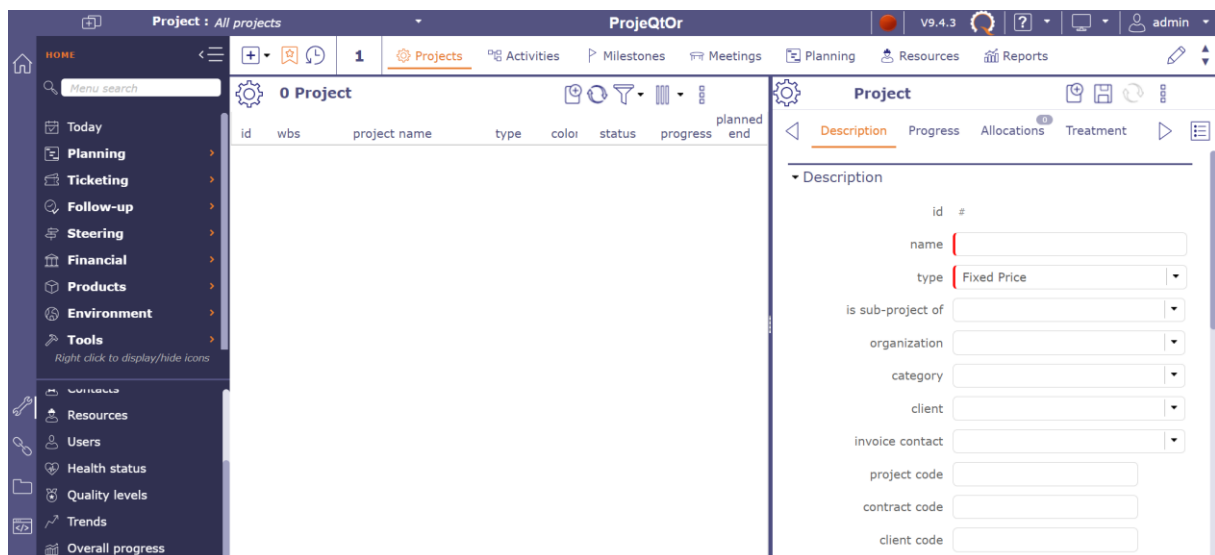
9. Si tous les diagnostics sont au vert, poursuivez en cliquant sur **continuer**. Le formulaire d'authentification de ProjeQtOr est affiché comme suit :

The image shows the ProjeQtOr login interface. It features the ProjeQtOr logo at the top. Below the logo are input fields for 'login' and 'password'. There is a 'remember me' checkbox and two buttons: 'OK' and 'Change password'. A red banner at the bottom states 'Upgrade in progress. Only admin user can connect.' A small warning icon is visible on the left side of the login area.

Vous pouvez alors achever l'installation en vous authentifiant pour la première fois en tant qu'administrateur (login=**admin** / password=**admin**).

Après vous être connecté à nouveau, vous arrivez sur un écran dans lequel vous devez choisir les modules à activer. **Laissez le choix par défaut** et sauvegardez les changements.

Enfin, vous voilà face au tableau de bord de l'application : l'installation est terminée.

The image displays the ProjeQtOr dashboard. The top navigation bar includes 'Project : All projects', 'ProjeQtOr', version 'V9.4.3', and a user profile 'admin'. The left sidebar contains a 'HOME' section with a menu search bar and a list of modules: Today, Planning, Ticketing, Follow-up, Steering, Financial, Products, Environment, Tools, and a 'PARAMETERS' section with Resources, Users, Health status, Quality levels, Trends, and Overall progress. The main area shows '0 Project' with a table header including id, wbs, project name, type, color, status, progress, and planned end. The right sidebar shows the 'Project' details for a selected project, with tabs for Description, Progress, Allocations, and Treatment. The 'Description' tab is active, showing fields for name, type (Fixed Price), is sub-project of, organization, category, client, invoice contact, project code, contract code, and client code.

10. Explorez les différentes fonctions. Familiarisez-vous notamment avec les éléments suivants :

- Le menu du profil utilisateur (admin) en haut à droite de l'écran qui contient les paramètres du profil connecté (donc le compte admin). Il vous permet notamment de modifier la langue de l'interface (pour mettre en français par exemple).
- Le menu **paramètres** (clé anglaise en bas à gauche) qui permet de définir les paramètres de base de vos projets.

ProjeQtOr est **un outil très riche**. Il nécessite une **configuration des types et des valeurs** pour pouvoir être utilisé dans le cadre d'un projet spécifique. Dans ce TP nous n'utiliserons qu'une partie limitée de ses fonctionnalités. Nous choisirons également d'effectuer une configuration simple.

V. Configuration de base

11. Commencez par **créer un client** (celui pour lequel vous travaillez) dans le menu **Environnement / clients**. Donnez juste son nom et son type (Customer).
12. Vous devez ensuite **créer des ressources**. Ce sont les personnes qui vont participer aux tests de validation fonctionnelle. Voici la liste de vos participants :

| Nom | Prénom | Rôle | Equipe | Entreprise |
|----------------|-------------|----------------|-------------|---------------------|
| <Votre prénom> | <votre nom> | Chef de projet | MOA | WonderWare |
| Annie | Peurien | Testeur | MOA | WonderWare |
| Yves | Oulaipas | Testeur | MOA | WonderWare |
| Joe | Black | Testeur | Client | Sunny Entertainment |
| Helen | Cooper | Testeur | Client | Sunny Entertainment |
| Sylvia | Smith | Testeur | Client | Sunny Entertainment |
| Gislaine | Deroche | | Développeur | WonderWare |
| Ulrich | Hochet | | Développeur | WonderWare |
| Rachid | Huilda | | Graphiste | WonderWare |

Vous devrez les créer dans le menu **Environnement / ressources**. Pour l'instant, remplissez les informations minimales, en sachant que vous pourrez toujours les modifier ou compléter au besoin.

Pour que les participants aux tests puissent accéder aux tickets et les gérer (créer, modifier, commenter, etc), vous devez obligatoirement leur donner un nom d'utilisateur (login) et cocher la case **est un utilisateur**.

13. Vous créerez également les quatre équipes d'utilisateurs. Pour cela, allez dans **Environnement/ Equipes**.

Vous pouvez affecter vos utilisateurs dans les équipes dans le menu **Environnement / Ressources**.

L'équipe MOA et les clients auront pour mission de tester l'application en suivant les scénarios de tests.

Les développeurs et le graphiste auront pour mission de corriger les problèmes rencontrés.

Paramétrage du projet

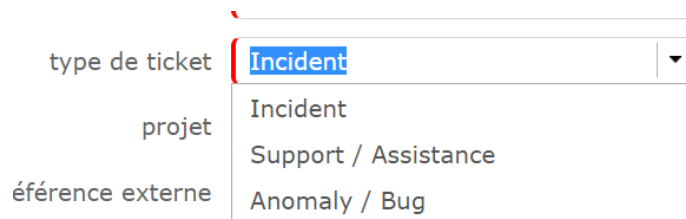
14. Créez un projet **Secret Things** dans le menu **Suivi / Projets**. Laissez le type **Fixed Price** par défaut. Sélectionnez le client. Donnez une description et un objectif.
15. Affectez vos utilisateurs au projet Secret Things dans le menu **Environnement / Affectations**. Ceux-ci seront affectés à 100% aux tests de validation fonctionnelle.

Paramétrage des types de tickets

Les testeurs vont remonter des situations qu'ils percevront comment des anomalies. Ces situations peuvent être de plusieurs types : des incompréhensions sur le fonctionnement attendu, des plantages du logiciel, des bugs, des situations inattendues, etc. Il faut donc définir une classification précise des différents cas afin de pouvoir les traiter selon la procédure adéquate.

Par exemple, un plantage logiciel devra être immédiatement analysée par le développeur, alors qu'une incompréhension sur le fonctionnement devra être discutée entre la MOA et le client afin de définir l'action à mettre en œuvre (ne rien faire, modifier le comportement du logiciel, apporter une information à l'utilisateur, etc).

Comme vous pouvez le constater, les types de tickets sont en nombre restreint par défaut :



The image shows a web form with three labels: 'type de ticket', 'projet', and 'référence externe'. The 'type de ticket' dropdown menu is open, showing three options: 'Incident' (highlighted in blue), 'Support / Assistance', and 'Anomaly / Bug'.

Les tickets d'incident et de support ne concernent pas directement la phase de tests fonctionnels. Il ne reste que le **bug** qui pourrait nous concerner, mais c'est une notion beaucoup trop vague.

16. Nous allons donc définir les nouveaux types de tickets suivants.

| Type | Description |
|-------------------|---|
| Observation | Ticket pour lequel on décide de ne réaliser aucune étude dans l'immédiat. |
| Dysfonctionnement | Problème constaté qui ne s'appuie sur aucune spécification mais dont la correction est nécessaire pour le fonctionnement normal du logiciel (bug, plantage, ...). |
| Non-conformité | Problème constaté de non-conformité par rapport aux spécifications. |
| Evolution | Problème représentant un écart par rapport aux spécifications pour lequel une future évolution du logiciel est nécessaire. |











Dans le menu **Ticketing**, cliquez sur **Tickets** puis **Types de tickets** pour les modifier.

Paramétrage des niveaux d'urgence des tickets

Certains problèmes constatés doivent être résolus plus rapidement que d'autres. Par exemple un problème qui empêche la poursuite des tests est à corriger avant une anomalie mineure dans l'affichage d'une page.

Par ailleurs, les développeurs qui travaillent à la résolution des problèmes n'ont pas une journée extensible. Il est donc nécessaire de classifier les tickets selon un ordre de priorité défini par le niveau d'urgence.

Par défaut, ProjeQtOr propose les niveaux d'urgence suivants :

|  3 Urgences      clos  | | | | | |
|--|------------|--------|---|-------|--------------------------|
| id | nom | valeur | couleur | ordre | clos |
| 3 | Not urgent | 1 |  | 10 | <input type="checkbox"/> |
| 2 | Urgent | 2 |  | 20 | <input type="checkbox"/> |
| 1 | Blocking | 4 |  | 30 | <input type="checkbox"/> |

Vous pouvez y accéder dans le menu **Ticketing** puis en sélectionnant **Urgences**. Cette classification est un peu trop simpliste pour gérer correctement les priorités de résolution des tickets.

17. Modifiez la classification comme suit :

| Nom | Signification |
|-------------|--|
| Bloquant | Problème à résoudre immédiatement |
| Très urgent | Problème à résoudre en priorité |
| Urgent | Problème à résoudre dès que possible |
| Non urgent | Problème dont la résolution peut attendre, pour l'instant |
| Mineur | Problème dont la résolution n'est pas essentielle (pouvant éventuellement n'être pas résolu du tout) |
| Non défini | Problème dont on n'a pas encore défini le niveau d'urgence. |

VI. Organiser des projets de test dans ProjeQtOr

Importer les plans de test dans ProjeQtOr

Comment les testeurs vont-ils tester le jeu ? Evidemment, Il n'est pas question de les laisser faire n'importe quoi. Souvenons-nous que l'objectif des tests de validation de fonctionnelle est de **démontrer que la solution logicielle répond au besoin qui a été exprimé**. En d'autres termes, on va vérifier que les fonctionnalités développées correspondent à ce qui a été demandé dans le **cahier des charges**.

A la fin de la séance de tests qui va durer deux semaines, on va décider si c'est le cas ou pas. On prononcera soit un **GO** (feu vert pour sortir la version en production) soit un **NOGO** (on repart en développement et on travaille sur une nouvelle révision du code).

Pour faire cette vérification, on va s'appuyer sur un **plan de de tests**. Celui-ci contient des **cas de test**. Chaque cas de test contient un scénario simple à tester. Le testeur va suivre le scénario et exécuter les opérations demandées. S'il rencontre un problème, il crée un ticket. Le cas de test est considéré comme terminé quand il a été exécuté et qu'aucun ticket n'est resté ouvert dessus.

Le plan de test est habituellement rédigé par la MOA. Il doit couvrir tous les besoins fonctionnels exprimés dans le cahier des charges.

Pour le projet Secret Things, le plan de tests a été rédigés dans un fichier excel accompagnant ce TP.

Le fichier ne contient qu'un petit échantillon de cas de test. Dans un vrai projet, celui-ci peut contenir des centaines de cas de test. Le plan de test doit maintenant être importés dans ProjeQtOr.

18. Cliquez sur le menu **Outils / Importer des données** puis importez le fichier des cas de test fournis avec ce TP. N'oubliez pas de sélectionner **cas de test** dans le menu déroulant **type d'éléments à importer**. Il se peut que vous soyez obligé de manipuler un peu le fichier pour qu'il puisse être chargé dans ProjeQtOr.

Définir le produit à tester

L'outil ProjeQtOr permet de gérer très finement les produits, version de produits ainsi que leurs composants.

Dans le cas présent, le produit testé est une plateforme web pour des jeux vidéo. Nous travaillons sur la version 1.0 de ce produit. Nous choisissons arbitrairement le ST1 (version 1 de la plate-forme web du jeu Secret Things) pour cette version.

19. Créer un produit dans le menu **Produits /Produits**.
20. Créer ensuite une version du produit dans le menu **Produits / versions de produit**. Appeler cette version **ST1**. Il s'agit d'une version majeure.

Définir les composants du produit

Un produit qu'il soit logiciel ou matériel, est généralement constitué de plusieurs composants. Dans le cas de notre plateforme Web, pour une question de simplicité, nous allons nous limiter à considérer qu'elle n'a que trois composants :

- Le module de gestion des utilisateurs (UTI) permettant à l'utilisateur de se connecter en ligne sur Internet et qui lui permettra d'accéder au jeu, de voir les nouveautés et les extensions proposées, de voir ses amis qui seront en ligne, etc.
- Le module permettant d'organiser des parties multijoueur (MUL).
- Le module permettant de faire des achats en ligne (PUR).

Nous considérons qu'il n'existe qu'une version de chacun de ces modules (UTI1, MUL1, PUR1).

21. Vous en savez assez pour **créer les composants et versions de composants** dans le menu **Produits**.

Définir la plateforme de test

Gérer les terminaux de test : les contextes

On entend par "terminal de test" l'appareil sur lequel le test est effectué. Le rendu peut être complètement différent d'un terminal à l'autre, d'une version d'OS à l'autre, d'une résolution d'écran à l'autre, etc.

Vous devez vous assurer que les contextes sur lesquels va tourner la plateforme sont bien définis dans ProjeQtOr. Vous pouvez les définir dans le menu **Environnement / Contextes**.

Pour rester simple, nous considérons que la plateforme web n'est prévue pour tourner que sur les PC présents dans la salle de cours et dans une seule résolution d'écran (pas de version mobile).

22. Ajoutez le système d'exploitation Windows 10 ainsi que les navigateurs présents sur le PC s'ils sont manquants.

Le référentiel documentaire

L'activité de test demande un savoir-faire et un travail conséquent en matière de documentation. Un projet peut générer plusieurs dizaines, voire centaines de documents de tous types et de toutes tailles.

Un outil de gestion documentaire s'avère donc très précieux pour faciliter ce travail. ProjeQtOr permet de le faire.

Le référentiel de documents s'appuie sur deux notions : les **types de documents** (comme le plan de test que nous avons déjà vu), et les **répertoires de documents**.

Les documents sont accessibles depuis le menu **Documents** de la barre d'outils à gauche de l'écran. En revanche, pour gérer les répertoires de documents, il faut aller dans le menu **Environnement /répertoires de documents**.

Par défaut, ProjeQtOr définit une hiérarchie comme suit :

| id | emplacement | nom | projet | produit du composant | clos |
|----|------------------------|---------------|--------|----------------------|--------------------------|
| 2 | /Product | Product | | | <input type="checkbox"/> |
| 5 | /Product/Conception | Conception | | | <input type="checkbox"/> |
| 8 | /Product/Deployment | Deployment | | | <input type="checkbox"/> |
| 9 | /Product/Exploitation | Exploitation | | | <input type="checkbox"/> |
| 3 | /Product/Need | Need | | | <input type="checkbox"/> |
| 4 | /Product/Specification | Specification | | | <input type="checkbox"/> |
| 7 | /Product/Testing | Testing | | | <input type="checkbox"/> |
| 1 | /Project | Project | | | <input type="checkbox"/> |
| 10 | /Project/Contract | Contract | | | <input type="checkbox"/> |
| 14 | /Project/Financial | Financial | | | <input type="checkbox"/> |
| 13 | /Project/Follow-up | Follow-up | | | <input type="checkbox"/> |
| 11 | /Project/Management | Management | | | <input type="checkbox"/> |
| 12 | /Project/Reviews | Reviews | | | <input type="checkbox"/> |

Vous voyez qu'il n'y a pas de dossier spécifique à un projet donné dans ProjeQtOr par défaut. Si on souhaite une telle organisation, c'est à nous-même de la définir.

23. Heureusement, pour ce TP nous ne gérons **qu'un seul projet et un seul document**, le plan de test (ou test plan). Je vous propose donc de créer un nouveau dossier à la racine portant le nom du projet, comme ci-dessous.

The screenshot shows the 'Répertoires de documents' window with a table of existing repositories. A new repository, '15 /Secret Things', has been added at the bottom. To the right, the 'Description' form for this repository is visible, showing fields for 'nom' (Secret Things), 'répertoire', 'emplacement' (/Secret Things), 'projet' (Secret Things), 'produit' (Secret Things), 'responsable' (Paul Ribeiro), and 'type par défaut' (Test Plan).

| id | emplacement | nom | projet | produit | clos |
|----|------------------------|---------------|---------------|---------------|--------------------------|
| 2 | /Product | Product | | | <input type="checkbox"/> |
| 5 | /Product/Conception | Conception | | | <input type="checkbox"/> |
| 8 | /Product/Deployment | Deployment | | | <input type="checkbox"/> |
| 9 | /Product/Exploitation | Exploitation | | | <input type="checkbox"/> |
| 3 | /Product/Need | Need | | | <input type="checkbox"/> |
| 4 | /Product/Specification | Specification | | | <input type="checkbox"/> |
| 7 | /Product/Testing | Testing | | | <input type="checkbox"/> |
| 1 | /Project | Project | | | <input type="checkbox"/> |
| 10 | /Project/Contract | Contract | | | <input type="checkbox"/> |
| 14 | /Project/Financial | Financial | | | <input type="checkbox"/> |
| 13 | /Project/Follow-up | Follow-up | | | <input type="checkbox"/> |
| 11 | /Project/Management | Management | | | <input type="checkbox"/> |
| 12 | /Project/Reviews | Reviews | | | <input type="checkbox"/> |
| 15 | /Secret Things | Secret Things | Secret Things | Secret Things | <input type="checkbox"/> |

Mettez-vous en tant que responsable du dossier.

24. Allez ensuite dans le menu **Documents** et ajoutez le plan de test dans le dossier **Secret Things**.

VII. Définir un projet de tests

Introduction et prérequis

ProjeQtOr propose une solution intégrée de gestion de projet offrant simultanément :

- la gestion de la charge et des ressources humaines et matérielles ;
- la gestion des contrats, de la facturation, des notes de frais ;
- la gestion opérationnelle des tâches du projet et de rédaction des livrables (réunions, documents...).

Nous n'allons en exploiter qu'une petite partie en nous cantonnant à un projet de test simple.

25. Avant d'aller plus en avant, assurez-vous que le chef de projet que vous avez créé (normalement c'est vous), est affecté au projet Secret Things en tant que chef de projet. Vous pouvez le faire dans le menu **Environnement /affectations**.
26. Vous devez également générer un mot de passe pour le chef de projet car cela n'a pas été fait jusqu'à présent. En tant qu'admin, allez dans le menu **Environnement / Utilisateurs**. Sélectionnez le chef de projet dans la liste, puis dans la fenêtre de droite, allez dans l'onglet **Détail** pour définir un mot de passe.

| id | nom de l'utilisateur | photo | profil | nom usuel | initiales | est une ressc | est un cont | vient de Ldap | clos |
|----|----------------------|-------|-------------------------|------------------|-----------|-------------------------------------|--------------------------|--------------------------|--------------------------|
| 11 | pribeiro | P | Chef de Projet | Paul Ribeiro | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 10 | rhuida | R | Membre d'équipe | Rachid Huïda | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 9 | uhochet | U | Membre d'équipe | Ulrich Hochet | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 8 | gderoche | G | Membre d'équipe externe | Gislaine Deroche | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 7 | ssmith | S | Membre d'équipe externe | Sylvia Smith | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 6 | hcooper | H | Membre d'équipe externe | Helen Cooper | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 5 | jblack | J | Membre d'équipe externe | Joe Black | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 4 | youlaipas | Y | Membre d'équipe | Yves Oulaipas | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 3 | apeurien | A | Membre d'équipe | Annie Peurien | | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 1 | admin | A | Administrateur | | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Utilisateur #11
Description | Affectations | **Détail**
Equipements attribués (0)
Type | Equipement | Marque | Modèle
Divers
mot de passe:
.....

vient de Ldap: ☐
ne reçoit pas les emails destinés à l'équipe projet: ☐
clé API: 8a50f2592a388fea1400c91c06f33098

Utiliser la gestion de projet

27. Connectez-vous maintenant avec le compte du chef de projet.

Vous voyez ce qu'un chef de projet a le droit de faire lorsqu'il est au moins affecté à un projet dans ProjeQtOr.

Nous nous focaliserons dans les sections suivantes sur les deux principales : la définition de l'**activité** (tâches et jalons) et les **affectations**. Mais voyons déjà les données d'un projet :

Project #1 - Secret... recorded 02/28 2022 02/24 2022 A

Description Progress Allocations Treatment Configuration Dependencies Links Files Notes

Products linked to this project

| product | start date | end date | closed |
|---------|------------|----------|--------|
| | | | |

Versions linked to this project

| version | start date | end date | closed |
|---------|------------|----------|--------|
| | | | |

Un projet se compose de plusieurs blocs de données :

- Description ;
- Progress (**Avancement**) ;
- Allocations (**Affectations de ressources**) ;
- Etc.

Vous retrouvez dans le bloc **Description** les informations que vous avez déjà entrées précédemment.

L'**état** du projet (en haut d'écran) indique la situation dans laquelle l'objet se trouve. **recorded** signifie enregistré, c'est-à-dire brouillon. Passez-le immédiatement en **in progress**.

Project #1 - Secret... in progress 02/28 2022 02/24 2022 A

Description Progress Allocations Treatment Configuration Dependencies Links Files Notes

Dans le bloc Allocation, vous devez retrouver l'équipe projet que vous avez déjà définie.

Project #1 - Secret... in progress 02/28 2022 02/24 2022 A

Description Progress Allocations Treatment Configuration Dependencies Links Files Notes

Allocations to projects 9

| id | resource | profile | start date | end date | rate (%) |
|----|------------------|----------------|------------|----------|----------|
| 3 | Annie Peurien | Project Member | | | 100 |
| 8 | Gislaine Deroche | Project Member | | | 100 |
| 6 | Helen Cooper | Project Member | | | 100 |
| 5 | Joe Black | Project Member | | | 100 |
| 11 | Paul Ribeiro | Project Leader | | | 100 |
| 10 | Rachid Hilda | Project Member | | | 100 |
| 7 | Sylvia Smith | Project Member | | | 100 |
| 9 | Ulrich Hochet | Project Member | | | 100 |
| 4 | Yves Oulaipas | Project Member | | | 100 |



Dans le bloc **Configuration**, liez le projet au produit et à la version sur laquelle nous travaillons.

Project #1 - Secret... in progress



02/28 2022 02/24 2022 A

Description Progress Allocations 9 Treatment Configuration Dependencies Links 0 Files 0 Notes 0

▼ Products linked to this project

| | product | start date | end date | closed |
|---|---------------|------------|----------|--------------------------|
|   | Secret Things | | | <input type="checkbox"/> |

▼ Versions linked to this project 1

| | version | start date | end date | closed |
|---|---------|------------|----------|--------------------------|
|   | ST1 | | | <input type="checkbox"/> |

Les activités et les jalons

Une fois que des ressources sont affectées à un projet, elles ont seulement le droit de se connecter à ProjeQtOr et de voir le projet dans leur tableau de bord : aucune tâche n'est assignée à qui que ce soit pour l'instant.

Votre travail de chef de projet de test consiste alors à définir les activités et jalons du projet. Dans notre cas, nous décidons que notre projet n'a qu'une seule phase : la phase de **Validation Fonctionnelle**. Nous allons donc créer des activités qui appartiennent toutes à cette phase.

28. Allez dans le menu **Planning / Activités**. Vous êtes alors face à la liste des tâches des projets (vide si c'est la première fois que vous venez là). Créez les activités de test comme ci-dessous.

| 3 Activités | | | | | | | |
|-------------|---------------|------------------|--------------------------------------|--------------------|----------|------------|-------------|
| | | | | | | | |
| id | projet | type d'activité | nom | échéance planifiée | état | avancement | responsable |
| 1 | Secret Things | Test fonctionnel | Tester le module de gestion | | recorded | 0 % | |
| 3 | Secret Things | Test fonctionnel | Tester le module de jeu multijoueurs | | recorded | 0 % | |
| 4 | Secret Things | Test fonctionnel | Tester le module d'achats en ligne | | recorded | 0 % | |

Définissez également des Jalons (menu **Jalons** en haut d'écran) pour vos tests.

29. Définissez alors à minima les six jalons suivants :

- Le jalon **Démarrage** (de type **Incoming**), que vous devez indiquer comme **jalon fixe** dans le bloc **Avancement**.
- Le jalon **Livraison des cas de test**, de type livraison (**deliverable**), que vous devez indiquer comme jalon fixe dans le bloc **Avancement**.
- Le jalon **Livraison Application** du projet de test, de type livraison (**deliverable**). Vous devez impérativement l'indiquer comme jalon fixe dans le bloc **Avancement**. Vous attendez ce prérequis pour faire face à une montée en charge brutale sur votre projet : l'exécution des tests.
- Le jalon **Acceptation Application**, de type date clé (**key-date**), que vous devez indiquer comme jalon fixe dans le bloc **Avancement**. Dans les faits, ce jalon pourrait être mobile. Dans la pratique, il suit immédiatement les tests de validation fonctionnelle.

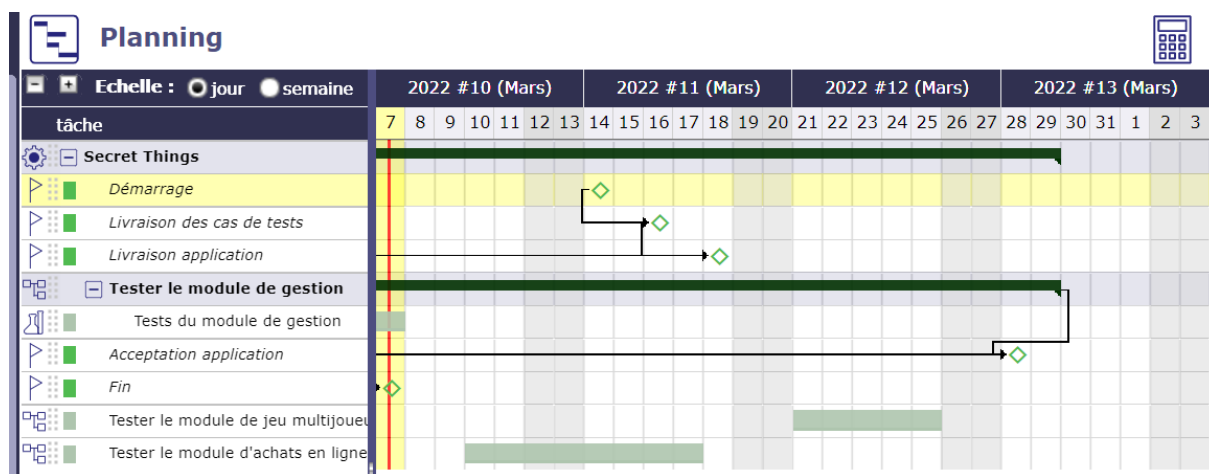
- Le jalon **Fin** du projet de test, de type livraison (**deliverable**), que vous devez indiquer comme jalon flottant dans le bloc **Avancement**.

Pour les jalons déclarés fixes, la date validée sera exigée. Commencez déjà à définir ici les dates de vos jalons même si elles sont fantaisistes : vous les stabiliserez ultérieurement.

- Une fois les phases et jalons définis, saisissez leur ordre de dépendance dans le temps dans le bloc **Dépendances**.

Vous obtenez ainsi la structure de votre planning de test : vous pouvez maintenant générer votre premier **diagramme de Gantt** !

- Cliquez sur le menu **Planning / Planning** pour voir le planning des tests. Vous obtenez un diagramme de Gantt qui devrait ressembler à peu près à celui-ci :



Par défaut, les lignes du Gantt sont ordonnées dans l'ordre de création des jalons et des activités. Cet ordre n'a pas forcément une signification. Avec la souris, ordonnez les lignes par cliquer-glisser. Une zone de saisie est prévue à cet effet à gauche de l'icône des activités et des jalons.

Il ne reste plus qu'à assigner les ressources aux trois activités de test. Répartissez les testeurs entre les deux activités en sélectionnant au moins un MOA et un client pour chaque activité.

Nous n'allons pas gérer ici les charges et les coûts. On considère que tous les testeurs sont disponibles à 100% pendant toute la durée des tests.

IX. Organiser les tests dans ProjeQtOr

Définir les cas de test

Les cas de tests ont déjà été chargés dans ProjeQtOr à une étape précédente. Vous pouvez les voir en allant dans le menu **Pilotage / Exigences / cas de tests**. Cependant, rien de nous empêche d'en ajouter.

32. Ajoutez le cas de test suivant :

Titre : *se déconnecter de l'espace personnel*

Description :

1 Cliquez sur l'icône « profil » en haut droite de l'écran.

2. Sélectionner l'option « déconnexion ».

33. N'oubliez pas de le lier au produit Secret Things et à la version ST1 dans l'onglet **Configuration**

Préparer les sessions des cas de test

La préparation des tests s'achève par la construction des séquences à dérouler, le plan d'exécution des tests, ainsi que l'affectation de ces séquences à l'équipe des testeurs. Les objets **Sessions** de ProjeQtOr ont la charge d'assumer ce rôle.

Créer une session : regrouper des cas de test

Une session peut se définir comme un regroupement ordonné d'instances de cas de test, un déroulement à exécuter ou en cours d'exécution (ou achevé).

34. Créez une session de test via le menu **Planning / Sessions de test**.

Dans le bloc **Description**, décrivez explicitement la session de test, par exemple "Tests fonctionnels UTI sous Firefox 97" en considérant que cette session regroupe tous les tests d'un type donné pour une configuration matérielle donnée.

Sélectionnez le type de session, le projet, le produit ou le composant, la version du produit ou du composant.

En général, on considère que les testeurs changent peu de configuration matérielle car ils ne disposent pas de configurations multiples de postes de travail. C'est pourquoi il est logique de définir des sessions en fonction du type de terminal et de la configuration matérielle. Dans notre cas nous avons même une configuration quasi unique dans la salle de cours.

Dans le bloc **Traitement**, choisir l'activité parente à laquelle est liée la session de test ainsi que le responsable de la session. Cette personne qui pourra modifier le statut d'avancement des cas de test, même si ce n'est pas elle qui les a réalisés.

Dans l'onglet **Avancement**, affectez les ressources à la session de test.

Dans l'onglet **Couverture**, sélectionnez les cas de test à réaliser dans cette session.

Définissez ainsi plusieurs sessions de test afin d'occuper au maximum le test de vos testeurs.

Définir l'enchaînement des sessions, les regrouper


Il est possible d'enchaîner les sessions de test en définissant des session parentes (des sessions contenant des sessions), des prédécesseurs et des successeurs.

Essayez d'organiser vos sessions d'une façon logique.

X. Exécuter des tests avec ProjeQtOr

35. Exécutez les tests comme décrit ci-dessous :

- Connectez-vous avec un profil testeur ;
- Sélectionnez le projet de test comme filtre principal (juste au-dessus du logo de l'application) ;
- Allez dans les sessions via le menu **Planning / Test sessions** ;
- Sélectionnez la session sur laquelle vous devez travailler.
- Dans l'onglet **Coverage**, sélectionnez le cas de test que vous voulez exécuter.
- Ici vous devriez normalement exécutez les tâches demandées, si la plateforme existait vraiment dans la vraie vie, mai en vrai, vous ne travaillez pas chez WonderWare et le jeu Secret Things n'existe pas.

Tout au long du déroulé des tests, les testeurs peuvent déclarer les anomalies en ouvrant des tickets. Pour cela, ils peuvent utiliser le raccourci  en haut de l'écran ou bien aller dans le menu **Ticketing /tickets**.


Pour montrer que vous avez bien compris le fonctionnement :

36. Créez deux tickets avec des profils de testeur. Faites travailler votre imagination pour trouver des anomalies fictives. Documentez vos tickets au maximum en y ajoutant des commentaires, documents, copies d'écran pouvant aider le développeur à la résolution du problème.
37. Terminez une session de test en mettant son statut à **Done**.

XI. Organiser la résolution des tickets

A l'issue d'une session de test, en tant que responsable du projet de tests, vous devez organiser une réunion de revue des tests. Le but de cette réunion est de décider des actions à mener concernant chaque ticket. On peut par exemple décider de corriger le logiciel dans sa version 1, d'attendre une future version pour le corriger, ou bien décider de ne rien faire du tout.

Les managers de WonderWare et du client peuvent être présents à cette réunion par visioconférence. Il faut donc leur présenter le plus clairement et simplement possible les résultats des tests et l'avancement des corrections.

38. Afin de préparer cette réunion, extrayez les tickets liés au projet (bouton  dans le menu **Ticketing /ticket**) et rédigez un diaporama (3 diapos maximum) contenant les informations nécessaires à la réunion.

