

Cours Logiciel Libre TP1

Gérer un projet

Halim Djerroud (hdd@ai.univ-paris8.fr)

1 Objectif du cours

1. Découper le projet en features
2. Mettre dans Backlog les features épiques
3. Affiner les features et créer des user storys
4. Mettre dans le features en ready
5. Création de comptes github
6. Initialisation du dépôt
7. Cloner le projet
8. Le workflow GIT à respecter
9. Faire le premier commit de sa branche
10. Merger les branches
11. Livrer un projet

2 Cahier des charges

AirPort est une compagnie qui gère un aéroport, cette dernière se voit attribuée un espace aérien à gérer d'un rayon de 100 KM au tour de l'aéroport, afin de se préparer AirPort fait appel à votre équipe pour réaliser un simulateur pour l'entraînement de son personnel.

AirPort souhaite avoir un simulateur proche du logiciel réel qu'elle va utiliser ultérieurement. Ce logiciel, très facile d'utilisation, il contient une seule fenêtre qui affiche des avions dans un rayon de 100 KM l'aéroport se trouve aux coordonnées suivantes (Latitude : 48.8333, Longitude : 2.6167). Chaque avion est affiché sous forme d'une croix avec le cap et l'altitude et la vitesse. À l'aide d'un simple clique sur un avion on peut lui envoyer un ordre de changer de cap ou d'altitude.

Avant de faire appel à vous la société AirPort a déjà commencée le projet avec une autre personne mais cet personne a malheureusement abandonnée le projet, mais une bonne partie de la conception a déjà été réalisée :

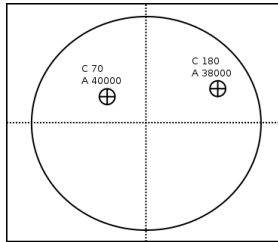


FIGURE 1 –

La solution est composée de deux applications, la première application permet de générer des avions aléatoirement qui traversent cet espace aérien (max 5 avions). Ces avions envoient chaque 15 secondes des informations (altitude, cap) à la deuxième application qui les affiche sur l'écran et grâce à un clic sur un avion un aiguilleur peut envoyer un ordre à l'avion, bien sûr l'application qui gère les avions doit prendre en compte ces ordres.

Note : unité de mesure le pied (ft), pour mesurer des distances verticales, le mille marin ou nautique, pour mesurer des distances horizontales,

Mission I :

Proposez un diagramme de classe et une implémentation en JAVA pour l'application qui gère l'affichage. Par souci de simplicité et de livrer rapidement un POC pour le client :

- l'affichage doit se faire sur une console sous forme (Numéro de vol, Latitude , Longitude, Cap, Vitesse, Altitude) l'affichage graphique sera traité dans une prochaine mission.
- Les informations qui proviennent des avions doivent être stocké en statique dans une matrice a deux dimensions.

Mission II :

Proposez un diagramme de classe pour le générateur d'avions. Proposez une implémentation en Java

Mission III :

Proposer un modèle de communication entre les avions et l'application d'affichage

Mission IV :

En Utilisant la bibliothèque SWING de java proposer une solution pour l'affichage

Annexe

Workflow GIT

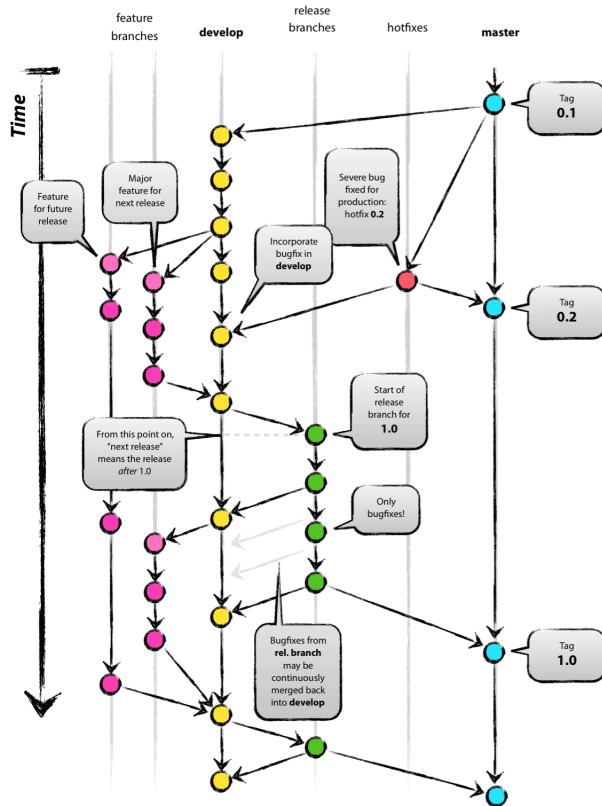


FIGURE 2 – Workflow git.

Source <https://nvie.com/posts/a-successful-git-branching-model/>