



Projet : les pirates

Etapes, travail à effectuer,
calendrier

Conseils de modélisation

Constitution des groupes

Modélisation et implémentation

Application des pirates.





Les pirates

- Le speech

Jack le Borgne et Bill Jambe-de-Bois sont deux pirates souhaitant prendre le commandement du navire à l'étendard noir, « Le Sanguinaire ». L'équipage doit donc les départager par un vote. Chacun des deux rivaux doit ainsi gagner en popularité (ou affaiblir son adversaire) afin de devenir le nouveau capitaine.





Le jeu des pirates

- Le Jeu des Pirates est un jeu de société de cartes dans lequel deux joueurs s'affrontent.
- Chaque joueur pioche quatre cartes. À tour de rôle, un pirate pioche une nouvelle carte et l'ajoute à sa main. Il doit ensuite décider s'il attaque son adversaire ou s'il s'attribue des points de popularité.
- Chaque pirate possède cinq cœurs rouges correspondant à ses points de vie. S'il n'a plus de cœur, alors il a perdu et son adversaire gagne.
- Le but du jeu est que son pirate atteigne cinq points de popularité ou simplement survive plus longtemps que son adversaire !



Zone de Popularité

Révolte
organisée

+1 Popularité

Zone d'attaque



Zone de Popularité

Main de Fer

+2 Popularité
-1 Vie

Abordage
Réussi

+1 Popularité

Au cours d'un abordage, le
pirate fait preuve d'une
grande bravoure et gagne deux
points de popularité



Discours
Inspirant

+1 Popularité

Main de Fer

+2 Popularité
-1 Vie

Révolte
organisée

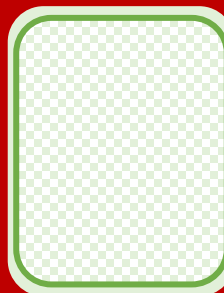
+1 Popularité

Coup de
sabre

-2 Vie

Abordage
Réussi

+2 Popularité





Le jeu des pirates

- Au début de son tour, le pirate pioche une carte et l'ajoute à sa main. Il choisit ensuite parmi sa main, composée de cinq cartes, une carte qu'il dépose :
 - Dans sa zone de popularité s'il s'agit d'une carte de popularité,
 - Dans la zone de son adversaire s'il s'agit d'une carte d'attaque.
- Dans un premier temps, vous implémenterez uniquement des cartes de popularité. Le premier pirate atteignant cinq points gagne.
- Dans un deuxième temps, vous complétez le jeu avec des cartes d'attaque, qui affecteront principalement les points de vie de l'adversaire.
- Enfin, dans un troisième temps, vous inventerez de nouvelles cartes avec d'autres effets, comme des cartes de défense, de sabotage, etc.



Etapes, travail à effectuer, calendrier

■ Séances restantes en POO :

- Séance 1 : CTD1 d'aujourd'hui (carte de popularité)
- Séance 2 : TP1 du 6 février
- Séance 3 : TP2 du 13 février
- Séance 4 : CTD3 du 17 février (autres cartes)
- Séance 5 : TP3 du 20 février
- Séance 6 : CTD4 du 4 mars (validation UML/groupe)
- Séance 7 : TP4 du 6 mars
- Séance 8 : TP5 du 13 mars (validation individuel)
- Séance 9 : CTD5 du 18 mars (travail UML individuel)

Le jeu que vous devez développer devra avoir ce comportement, mais sera une version console !

- Travail en groupe pour la partie conception,
- Travail individuel pour la partie développement,
- Validation,



Etapes, travail à effectuer, calendrier

- Séance 1 : CTD1 d'aujourd'hui
 - Définition des groupes
 - Travail sur le diagramme de séquence système,
 - Travail sur le diagramme de classes
- Séance 2 : TP1 du 6 février
 - Début du développement
 - Sous l'onglet « Les Pirates » sous Moodle cliquer sous :
« Lien projet git » et compléter comme indiqué le champ de texte
- Séance 3 : TP2 du 13 février
 - Fin développement de la première étape (carte popularité)

- Travail en groupe pour la partie conception,
- Travail individuel pour la partie développement,
- Validation,



Etapes, travail à effectuer, calendrier

- Séance 4 : CTD3 du 17 février
 - Travail sur les cartes spécifiques : compléter le diagramme de classes précédent,
 - Travail sur le (ou les) diagramme(s) de séquence détaillé(s)
 - Séance 5 : TP3 du 20 février
 - Séance 6 : CTD4 du 4 mars : présentation par groupe de la partie UML
 - Présentation notée, par groupe, de la partie UML
 - Séance 7 : TP4 du 6 mars
 - Séance 8 : TP5 du 13 mars
 - Recette : vérification fonctionnelle **individuelle**
 - Séance 9 : CTD5 du 18 mars fin du rapport individuel UML
 - Rendu individuel Lundi 24 mars avant 7H30
- Travail en groupe pour la partie conception,
 - Travail individuel pour la partie développement,
 - Validation,



Etapes, travail à effectuer, calendrier

- Exigences concernant votre travail (chaque point contribuera à l'évaluation) :
 - Séparer la partie modèle de la partie qui gère les affichages,
 - Utiliser toutes les techniques apprises durant les deux semestres : énumérations, tableaux, héritage, classes abstraites, interfaces, etc.,
 - Utiliser Git régulièrement,
 - Respecter les règles de SonarQube,
 - Appliquer les bonnes pratiques étudiées durant l'année (par exemple, utiliser la visibilité la plus adaptée aux attributs et aux méthodes),
 - Modéliser avec précision les diagrammes de classes et de séquence (exemple : l'implémentation de la multiplicité 0..1 est différente de celle de 1).



Etapes, travail à effectuer, calendrier

- **Rendu Lundi 24 mars avant 7H30** : sous la forme d'un dépôt git.
 - **Attention** tout push après 7H30 ne sera pas pris en compte !
 - En plus du code vous devrez pousser **un pdf** contenant une explication du fonctionnement de vos cartes spéciales et vos diagrammes UML :
 - Diagramme de séquence Système,
 - Diagramme de classes,
 - Diagramme de séquence détaillé d'une ou plusieurs cartes spéciales selon la difficulté (discuté avec votre enseignant durant les CTD3 et 4)
 - **Attention** les diagrammes devront **être lisibles une fois imprimés** donc :
 - Sérer les lignes de vie,
 - Utiliser le mode paysage si nécessaire,
 - Utiliser plusieurs pages si nécessaire en remplaçant les objets en haut de chaque page.
 - Un diagramme non lisible une fois imprimé vaudra 0.

Le pdf devra :

- Être **nommé avec votre nom**
ex : chaudet.pdf
- Contenir une page de garde,
- Les pages devront être numérotées



Conseils de modélisation

- Remarques issues du travail sur les Gaulois :
 - L'utilisation de l'héritage permet de simplifier le développement,
 - La présence de sorties textuelles dans l'ensemble du code complique l'utilisation du domaine métier.
- Dans ce projet, vous devrez concevoir une architecture séparant clairement les affichages du reste du développement. Vous devrez utiliser des interfaces pour faciliter cette séparation.
- Aucun texte ne devra provenir d'une classe non destinée à l'affichage. Par conséquent, évitez de créer des méthodes retournant des chaînes de caractères.
- Dans le développement logiciel, certaines architectures permettent d'organiser le code de cette manière, comme l'architecture ECB ou l'architecture MVC.



Constitution des groupes

- 6 groupes de **5 étudiants** :
 - G1 : PINARD Marius, JEANMART Théo, BOSSARD Alec, CARMOUZE Guilhem, APEDOH senam Amenyo Jules
 - G2 : HANNA Eliam, VIOLEAU Lylia, FRENOY Jean-Gabriel, JANATI Samy,
 - G3 : Louis CLOUET, Leo AHMED MUSHTAQ, Yoann SMETS, Martin LETEURTRE, Abdelhakim HAYATE
 - G4 : Adjil Aram SECK, Wissal MEHREZ , Hela BAKHTI
 - G5 : TAYOU MBEDE, Djibril BA, BARBARA, WACHILL Imran
 - G6 : MORALES Thomas, ESCAICH Loïc, IJERDAOUN Fairouz, MATHLOUTHI Rayen, HOUDASS Abdelbasset
 - G7 : WALI Wassim, PERRIN Alexandre, POMMEREAU Raphaël, TODESCHINI Alexandre, SICI Axel,
 - G8 : ANDRES Clément, DELHEM Sébastien, PAZZE Eliot, RUIMY Yaniv, TANIOS Antoine