

- **Respect des fonctionnalités demandées dans le cahier des charges.** Une absence (ou une contradiction) avec le cahier des charges donne forcément lieu à une sanction. **(Total 60%)**
  - Authentification et gestion des login (5%)
  - Découverte des indoor users (Decentralized fashion) (10%)
  - Découverte des outdoor users (Centralized Presence Server) (20%)
  - Envoi/Réception des messages entre utilisateurs (10%)
  - Gestion de l'historique (avec recherche et affichage) des anciens messages (15%)
- **Architecture et présentation du code (Total 20 %)**
  - Facilité de déploiement du système après un clone (e.g. maven ou autre) et facilité d'exécution et de configuration en se basant sur la doc fournie (fichiers readme, .dat, place holders, etc.) (5%)
  - Architecture des systèmes choisies et respect des specs (e.g. MVC pour le code, REST pour la partie Web, JavaBean pour la partie JDBC, FireManagement pour les entités SWING, etc.) (10%)
  - Respect des bonnes pratiques de la POO et de Java en particulier (organisation sous forme de packages pertinents, gestion des attributs et appels aux méthodes efficaces – e.g. le syndrome du private/static, e.g. gestion de la complexité - e.g. mémoire et heap space - soignée) (5%)
- **Manuscript (Total 20%)**
  - Pertinence du plan et du contenu (1%)
  - Motivations de choix (technos ou conceptuels) pris (5%)
  - Manuel d'utilisation du produit (10%)
  - Procédures de tests et de validation (4%)