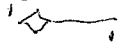
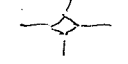


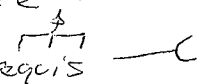
- Diag. Objet vs Diag Classe
- La multiplicité (lignes) est identique
  - Structure Objet est plus parlante
- Diag. Des Classes - Présentation
- Nom de classe
  - Attributs
  - Opérations (methodes)
  - Visibilité (+ public, - private, # protect)
  - Relations du Diag. De Classe
  - multiplicité (ex 1, 0..1, m..n)
  - Agrégation → couplage plus fort



- Association N-aire



- Dépendance
- Héritage
- Interface requis



- Généralisation du Diag Classes
- hiérarchie, Démarche ascendante
  - classes abstraites par nom en Italique
  - La composition est une agrégation forte
  - Les cycles de vie des éléments et de l'agrégat coïncident.
  - L'agrégation présente une vue subjective
  - Pour les interfaces on ajoute « Interface »
  - Interface symbolique
  - Les Packages sont représentés par
  - Relation entre les packages = dépendance

Les 2 Diagrammes d'interactions

- Le Diag. de collaboration
- Montre les interactions entre les objets
  - Montre les relations structurelles
- Les Composants du Diag Collaboration
- instances dans des rectangles
  - liens entre des objets
  - des messages qui déclenchent actions
  - Les messages doivent être numérotés [i := 1..n] msg

- envois conditionnel [x > y] : msg

Le Diagramme de Séquence

- Vue pour un scénario d'un UC
- Montre la communications entre objets
- Chaque objet est représenté par une barre verticale
- Le temps s'écoule du haut vers le bas
- Représenter le scénario = séquence d'événements
- types de messages : synchrones, asynchrones, réflexifs, récursifs, destructifs.

- Les formes du diag. de séquence représentent indirectement la structure : contrôle centralisé vs décentralisé

- Les Diag. d'états-transitions
- représentent le comportement d'une classe ou d'une collaboration.
  - concepts : automates, états, transitions, événement, gardes
  - Les automates représentent le comportement réactif obj.
  - Les automates et les scénarios sont complémentaires
  - un état = valeur des attributs + présence de liens
  - 2 cas particuliers : Début (●) et fin (⊙)
  - Peut avoir plusieurs fin, mais 1 seul début
  - Le passage d'un état à un autre s'exécute lorsqu'un événement est déclenché par un événement
  - Une garde = condition booléenne [ ] [ état ] [ ]
  - Notation (H) = mémorisation du dernier état
- Les Diagrammes D'activités
- Concept = Représentent le comportement des objets en terme d'activité.
  - Activité = stéréotype d'état
  - Concepts clés : garde, barre de synchro. swimlanes.

- Critères Ergonomiques (GUI)
- La compatibilité : réduire le transfert de connaissance
  - Le guidage possède 2 types : explicite et implicite
  - L'homogénéité rend le système prévisible, par ex. utiliser le même vocabulaire pour les commandes
  - La flexibilité permet la "paramétrisation"
  - Le traitement des erreurs inclus correction et protection

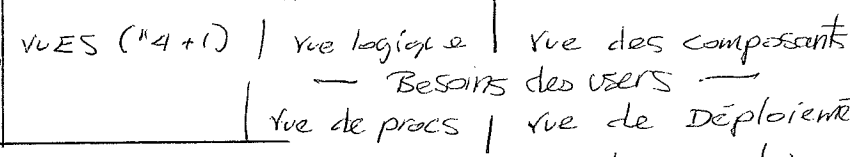
- L'écran.
- L'utilisateur parcourt l'écran en "I"
  - Le poignet déplace plus facilement de gauche à droite
  - La couleur ajoute info sans modifier de forme
  - Limiter le nombre de polices à 4 maximum.
  - Mise en évidence : Majuscule, encadrement, alignement
  - Les messages seront constructifs et non critiques
- Le Diagramme de Composants
- Décrire l'architecture physique et statique d'une application (fichiers sources, bibliothèques...)
  - Vue logique de l'environnement de dev.

Le Diagramme de déploiement

- Montre la disposition physique du matériel

Assurance de Qualité

- Analyse Statique = test boîte blanche (incluant test unitaire, test intégration)
- Analyse Dynamique = test boîte noire (incluant test de système, validations)



UP: Unified Process, depuis 1997 suite à UML 1.0 par Jacobson, Rumbaugh et Booch. Répond à Qui participe. Quoi produire. Comment réaliser et Quand. Introduit le principe des 4 "p" soit personnes (Architectes, développeur, testeur...), processus (patterns, domaines...), projet ou phase (du processus d'affaire au dev) et produit (livraison). UP est itératif et incrémental : décomposition en sous-projets. Cycle de répétition des phases de projet-innovation (visibilité) Élaborer la capacité de collaboration constructivité