


# LOG210 SÉANCE #13

1. Examen final  57.43m
2. Enquête sur la rémunération d'ingénieur
3. Résautage
4. Stackoverflow developer Survey
5. Liste de recruteurs
6. Sales statistics
7. 97 things
8. Exercices de préparation

async: 88m, sync: 92m

# EXAMEN FINAL

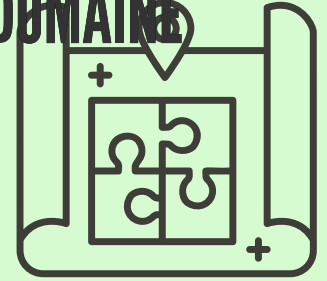
- Durée: 3 heures
- Nombre de questions: 83
- Documentation permise: Tout documentation écrite

# ANALYSE ET CONCEPTION ORIENTÉES OBJECT

- Développement itératif, évolutif et agile
- Processus unifié:
  - Phases
    - Inception
    - Élaboration
    - Construction
    - Transition
  - Disciplines

# ANALYSE

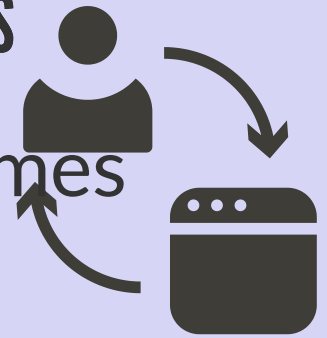
- Exigences
  - Cas d'utilisation
  - **FURPS**
- Diagramme de cas d'utilisation
- MDD
- Diagramme de séquence système
- **Diagramme d'activité**
- Contrats



- Utilité du MDD
- Concepts
  - identification par les noms
  - identification par listes de catégories
- Associations
  - critères pour des associations utiles
  - identification par listes de catégories
  - Noms, cardinalités, rôles
- Attributs
  - Simple vs complexe
  - Données primitives vs non primitives

# MAITRISER LA MÉTHODOLOGIE POUR CONSTRUIRE LES DSS

- Raison d'être des opération systèmes
- Relation avec le contrat
- Paramètres
- Relation avec les RDCU
- Retour d'opération  $\Leftrightarrow$  RDCU



# CHP14 DIAGRAMME D'INTERACTION

- Nommez moi les deux diagramme d'interaction?



# CHP14 DIAGRAMME D'INTERACTION



- Nommez moi les deux diagramme d'interaction?
  - Avantages/désavantages?



# CHP14 DIAGRAMME D'INTERACTION



- Nommez moi les deux diagramme d'interaction?
  - **Avantages/désavantages?**
- Un diagramme d'interaction est composé de classe ou d'instance de classe?

# CHP14 DIAGRAMME D'INTERACTION



- Nommez moi les deux diagramme d'interaction?
  - **Avantages/désavantages?**
- Un diagramme d'interaction est composé de classe ou d'instance de classe?
- Comment représente t-on un résultat de message en UML.

# CHP14 DIAGRAMME D'INTERACTION



- Nommez moi les deux diagramme d'interaction?
  - **Avantages/désavantages?**
- Un diagramme d'interaction est composé de classe ou d'instance de classe?
- Comment représente t-on un résultat de message en UML.
- Comment représente t-on une création d'instance en UML?

# CHP14 DIAGRAMME D'INTERACTION



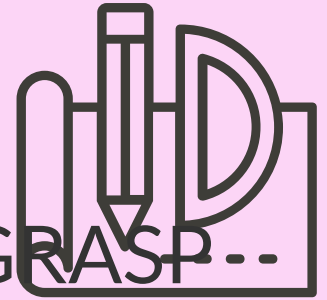
- Nommez moi les deux diagramme d'interaction?
  - **Avantages/désavantages?**
- Un diagramme d'interaction est composé de classe ou d'instance de classe?
- Comment représente t-on un résultat de message en UML.
- Comment représente t-on une création d'instance en UML?
- Pourquoi utilise t'on des cadres dans le diagramme d'interaction?



- Raison d'être
- Postconditions et leur relation avec le modèle du domaine
- Création d'instance et destruction
- Modification d'attribut
- Création et destruction d'associations
- Utiliser le langage du modèle du domaine!!!

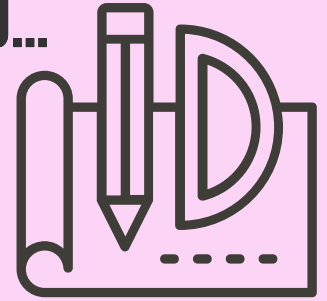
Created by Mohammed Ra

# CONCEPTION

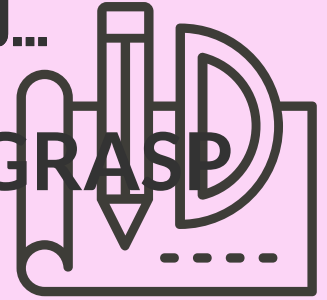


- RDCU: Diagramme d'interaction + GRASP...
- Interaction:
  - Séquence
  - Communication
- Diagramme d'état
- Diagramme de package
- Diagramme de déploiement
- Diagramme de composant

# MAITRISER LA MÉTHODOLOGIE POUR CONSTRUIRE LES RDCU...



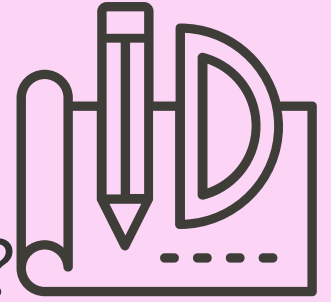
- Raison d'être de la réalisation de cas d'utilisation
- Faire le design à partir d'un contrat
- Création d'instance != Création d'association
- Traçabilité (lien avec les postconditions)
- Étapes à suivre
  - Choisir le contrôleur
  - Réaliser les postconditions du contrat en créant les méthodes...
- Où mettre les méthodes? Justifier avec les patrons GRASP
  - **Contrôleur**
  - **Créateur**
  - **Expert en information**
  - **Faible couplage**
  - **Forte cohésion**
  - **Polymorphisme**
  - **Protection de variation**
  - **Fabrication pure**
  - **Indirection**



- Être en mesure d'identifier les patrons GRASP dans un design
- De faire un design avec les patrons GRASP
- De critiquer un design par rapport à l'application des patrons
- De justifier correctement un patron GRASP

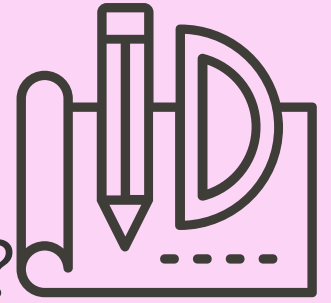


# CHP22 GRASP SUITE



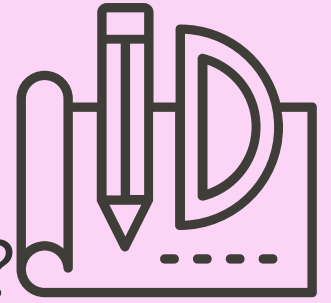
1. Quel sont les 4 derniers patrons GRASP?

# CHP22 GRASP SUITE



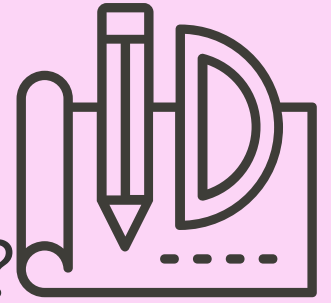
1. Quel sont les 4 derniers patrons GRASP?
2. Quel patron correspond à une invention de la part du concepteur?

# CHP22 GRASP SUITE



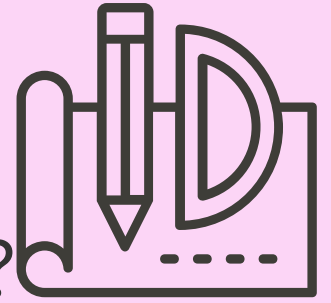
1. Quel sont les 4 derniers patrons GRASP?
2. Quel patron correspond à une invention de la part du concepteur?
3. Quel patron permet de réduire le couplage?

# CHP22 GRASP SUITE



1. Quel sont les 4 derniers patrons GRASP?
2. Quel patron correspond à une invention de la part du concepteur?
3. Quel patron permet de réduire le couplage?
4. Quel patron permet d'éliminer les effets indésirable des composants instable?

# CHP22 GRASP SUITE



1. Quel sont les 4 derniers patrons GRASP?
2. Quel patron correspond à une invention de la part du concepteur?
3. Quel patron permet de réduire le couplage?
4. Quel patron permet d'éliminer les effets indésirable des composants instable?
5. Quel patron permet le concept d'ouverture à l'extensibilité et fermeture aux modifications?

# AUTRE PATRON

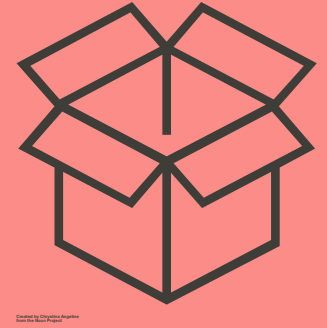
- Patron faire soi même

# CHP12 DIAGRAMME DE PACKAGE



1. Quels artéfacts influence le diagramme de package?
2. Qu'est ce que l'architecture logicielle?
3. Quel sont les couche typiques en OO?
4. Avantages de la modélisation en couche?

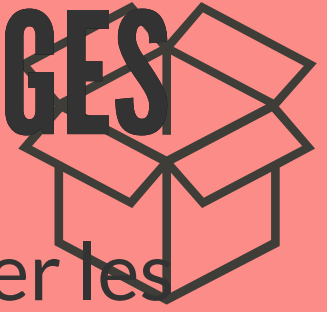
# ARCHITECTURE EN COUCHES



- Patrons architecturaux en couche
- Organisation en package
- Couplage entre package
- dépendances « ascendantes » interdites
- couches basses sont plus stables
- Comprendre MVC dans le contexte du patron architectural « layers »
- Séparation MVC



# CHP29 CONCEPTION DE PACKAGES

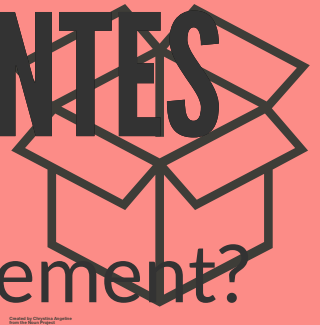


1. Quels principes utilise t-on pour organiser les packages?

# AFFINEMENT DU MDD

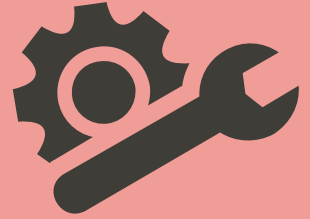
- Généralisation/spécialisation
- Hiérarchie (Service d'autorisation)
- Roles: Nom et multiplicité
- Éléments dérivés
- Packages et dépendances
- Role: Concepts vs association
- Composition vs aggrégation

# CHP31 DÉPLOIEMENT/COMPOSANTES



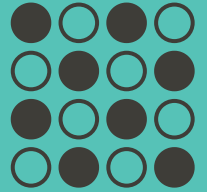
1. Quel est l'utilité du diagramme de déploiement?
2. Quel sont les type de nœuds dans un diagramme de déploiement?

# CHP20 OUTILS UML



1. Qu'est ce que la pro-ingénierie?
2. Qu'est ce que la rétro-ingénierie?
3. Qu'est ce que l'ingénierie cycle?
4. Comment choisi t'on un outils UML?

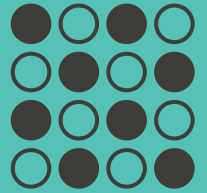
# GOF



Created by Jonathan Li  
from the Noun Project

- **Adaptateur**
- Commande
- Composite
- Template method
- **Décorateur**
- Fabrique Abstraite
- **Fabrique concrète**
- Itérateur
- **Proxy**
- Observateur
- Singleton
- Stratégie
- Visiteur

# CHP23 APPLICATION DES GOF



Created by Jonathan Li  
from the Noun Project

1. Quels patrons GRASP retrouve t'on dans plusieurs patrons GOF.
2. Faire la mise en correspondance entre les patrons GOF et les patrons GRASP.

# CHP30 APPLICATION GOF



Created by Jonathan Li  
from the Year 1 Project

1. Qu'est ce qui motive l'utilisation d'un patron GOF?
2. Qu'est ce que la paternite?

# FAUTE, ERREUR, ÉCHEC

1. Donnez moi un exemple de faute, erreur et échec.




# AUTRES

- Test et refactorisation
- TDD
- Notation UML
- **Dette techniques**
- Laboratoire

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS


1. Examen final
2. Enquête sur la rémunération d'ingénieur...  
31.02m
3. Résautage
4. Stackoverflow developer Survey
5. Liste de recruteurs
6. Sales statistics
7. 97 things
- ≡ 8. Exercices de préparation

# ENQUÊTE SUR LA RÉMUNÉRATION D'INGÉNIEUR

- <https://www.genium360.ca/fr-ca/services-aux-entreprises/enquete-sur-la-remuneration/>
- [https://drive.google.com/file/d/1UnInAz3y1it-I5S\\_ED2W-dj2JVJ8clTC/view?usp=sharing](https://drive.google.com/file/d/1UnInAz3y1it-I5S_ED2W-dj2JVJ8clTC/view?usp=sharing)

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS


1. Examen final
2. Enquête sur la rémunération d'ingénieur
3. Résautage 
4. Stackoverflow developer Survey
5. Liste de recruteurs
6. Sales statistics
7. 97 things
8. Exercices de préparation

# RÉSAUTAGE

[https://drive.google.com/open?  
id=1Cm2WTaG5KN0sqonAfOnbxjcuNfbMYFBW](https://drive.google.com/open?id=1Cm2WTaG5KN0sqonAfOnbxjcuNfbMYFBW)

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS


1. Examen final
2. Enquête sur la rémunération d'ingénieur
3. Résautage
4. Stackoverflow developer Survey 
5. Liste de recruteurs
6. Sales statistics
7. 97 things
8. Exercices de préparation

# DEVELOPER SURVEY RESULTS

- <https://insights.stackoverflow.com/survey/2019>
- <https://insights.stackoverflow.com/survey/2020>

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS

1. Examen final
2. Enquête sur la rémunération d'ingénieur
3. Résautage
4. Stackoverflow developer Survey
5. Liste de recruteurs 
6. Sales statistics
7. 97 things
8. Exercices de préparation




# LISTE DE RECRUTEURS

[https://drive.google.com/open?  
id=11XYE6k8m6YIcbAJ6hGBfkmIIJLH\\_DRTp](https://drive.google.com/open?id=11XYE6k8m6YIcbAJ6hGBfkmIIJLH_DRTp)

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS


1. Examen final
2. Enquête sur la rémunération d'ingénieur
3. Résautage
4. Stackoverflow developer Survey
5. Liste de recruteurs
6. Sales statistics 
7. 97 things
8. Exercices de préparation

# SALES STATISTICS

[https://drive.google.com/open?  
id=1jJT5YChfylxJNfSQQPH1IsBLab2CQYpG](https://drive.google.com/open?id=1jJT5YChfylxJNfSQQPH1IsBLab2CQYpG)

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS

1. Examen final
2. Enquête sur la rémunération d'ingénieur
3. Résautage
4. Stackoverflow developer Survey
5. Liste de recruteurs
6. Sales statistics
7. 97 things 
8. Exercices de préparation

# 97 THINGS

## 97-THINGS-EVERY-PROGRAMMER-SHOULD-KNOW

- <https://github.com/97-things/97-things-every-programmer-should-know/blob/master/en/SUMMARY.md>

## 97-THINGS-EVERY-AGILE-DEVELOPER-SHOULD-KNOW


- <https://github.com/97-things/97-things-every-agile-developer-should-know>

## 97-THINGS-EVERY-SOFTWARE-ARCHITECT-SHOULD-KNOW

- <https://github.com/97-things/97-things-every-software-architect-should-know>

# LOG210 SÉANCE #13

## ANALYSE ET CONCEPTION DE LOGICIELS

1. Examen final
2. Enquête sur la rémunération d'ingénieur
3. Résautage
4. Stackoverflow developer Survey
5. Liste de recruteurs
6. Sales statistics
7. 97 things
8. Exercices de préparation 

# LOG210

## EXERCICES EN LIGNE

<https://github.com/yvanross/LOG210-exercices>

# EXERCICES DIAGRAMME D'ACTIVITÉ



Par ordre de complexité

1. Retour de voiture louée
2. Processus d'achat sur le Web
3. Recette de cuisine
4. Ordinateur de plongée

Sans solution:

1. Demander un remplacement



# EXERCICES DIAGRAMME D'ÉTAT



Created by Rudez Studio  
from the Noun Project

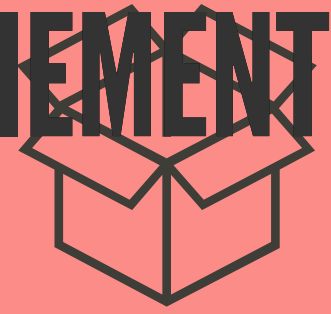
Par ordre de complexité

1. Téléphone intelligent
2. Guichet automatique
3. Vidéo projecteur

Autres exercices sans solution

1. CU29-Annuler un service
2. CU30-Confirmer une visite supervisée
3. CU31-Confirmer des échanges de garde
4. CU32-Rédiger une note d'observation
5. CU33-Corriger une note d'observation

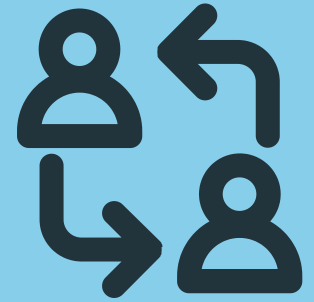
# EXERCICES DIAGRAMME DE DÉPLOIEMENT



1. Application web
2. Tutorial online
3. UML deployment diagram - Apple iTunes
4. UML Deployment Diagrams Examples

# SÉANCE #13

## RÉTROACTION: PAGE D'UNE MINUTE



Created by Prithvi  
from the Noun Project

1. Quels sont les deux [trois, quatre, cinq] plus importants [utiles, significatives, surprenantes, dérangementantes] choses que vous avez apprises au cours de cette session?
2. Quelle (s) question (s) reste (s) en tête dans votre esprit?
3. Y a-t-il quelque chose que tu n'as pas compris?

<https://1drv.ms/u/s!An6-F73ulxAOhVyiCB46jTeINVLs>



