

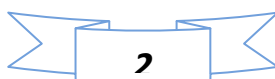
Projet Java :

Gestion informatique de l'emploi du temps d'une école



Sommaire

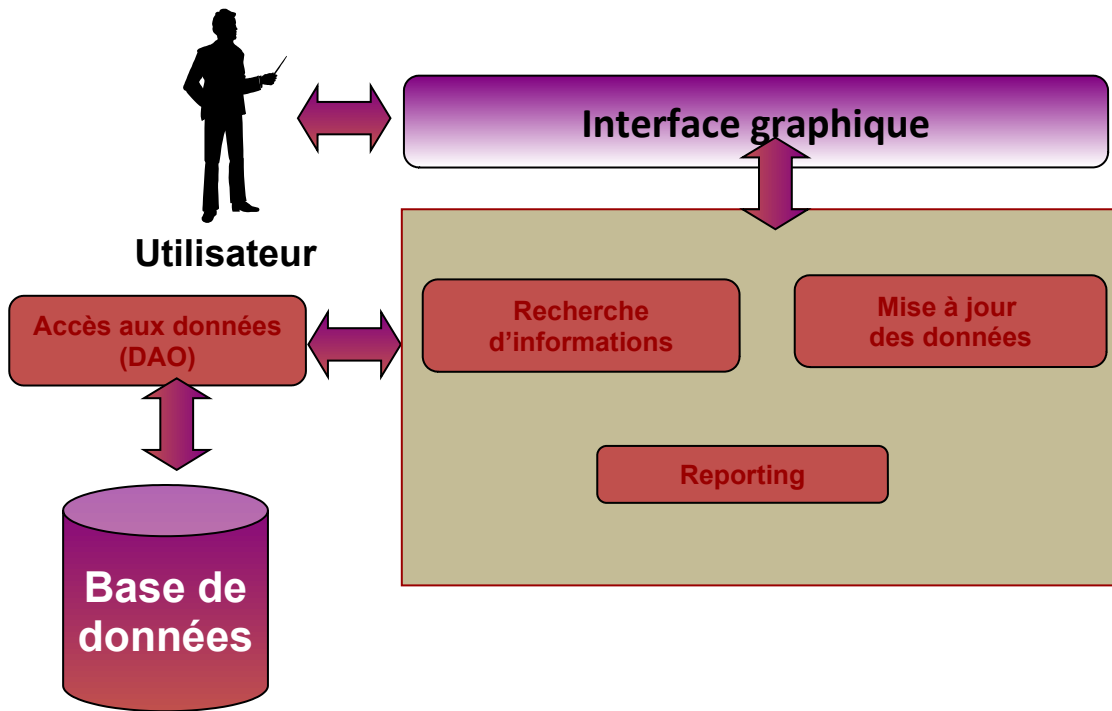
| | |
|--|----|
| Cahier des charges du projet..... | 3 |
| <i>Objectif</i> | 3 |
| <i>Architecture générale du système (pattern MVC)</i> | 3 |
| <i>Description du modèle relationnel</i> | 4 |
| <i>Création de la base de données (BDD)</i> | 5 |
| <i>Le module Recherche d'informations</i> | 5 |
| <i>Le module Mise à jour des données</i> | 7 |
| <i>Le module d'accès aux données (pattern DAO)</i> | 7 |
| <i>Le module de Reporting avec JFreeChart</i> | 7 |
| <i>Le module Interface graphique</i> | 8 |
| Contraintes techniques..... | 9 |
| Sources et ressources..... | 9 |
| Equipes et planning..... | 10 |
| Travail à faire et planning..... | 10 |
| <i>Phase de conception (patterns MVC, DAO et maquettes graphiques) AVANT la phase de réalisation</i> | 10 |
| <i>Livrables à déposer sur campus : PowerPoint et code</i> | 10 |
| Déroulement de la soutenance..... | 11 |
| Critères d'évaluation du travail à présenter..... | 12 |
| Recommandations..... | 12 |



Cahier des charges du projet

Objectif

Ce projet vise à concevoir un système de gestion de l'emploi du temps d'une école. L'architecture générale de ce système est décrite dans le schéma ci-dessous :



Architecture générale du système (pattern MVC)

Dans cette section, nous présentons l'Architecture générale de gestion de cet emploi du temps. Ce système compte principalement 5 modules :

- Le module de **Recherche d'informations** : toutes les consultations possibles de l'emploi du temps dans la base de données, selon plusieurs critères de recherche
- Le module de **Mise à jour des données** : toute opération de modification, ajout ou suppression dans l'emploi du temps
- Le module de **Reporting** : des statistiques sous forme de graphes (camemberts, histogrammes etc.)
- Le module **Accès aux données (DAO)** interroge ou met à jour les données de la BDD et communique avec les 3 modules précédents
- L'**Interface graphique** communique avec les 3 premiers modules pour visualiser l'emploi de temps

D'après le pattern **MVC**, votre interface graphique constitue la **Vue** (seulement l'affichage) dépendante des actions de l'utilisateur (gestion événementielle) au niveau **Contrôleur** (modules de **Recherche**, de **Mise à jour** et de **Reporting**). Ce dernier va demander au **Modele** de récupérer ou de mettre à jour - via le module d'**accès aux données (DAO)** - les informations de la base de données, de les organiser et de les assembler (par exemple, en les stockant dans des collections). Puis le **Contrôleur** va demander au **Modele** les données, les analyser, prendre des décisions et renvoyer le texte à afficher à la **Vue**.

Description du modèle relationnel

Dans cette section, nous vous donnons ci-dessous une description du modèle relationnel souhaité pour une bonne exploitation du système de gestion de l'emploi du temps de l'école. Il s'agit d'un modèle décrivant de façon réaliste les données de l'emploi du temps.

Remarques concernant le format des attributs des relations (futures tables) ci-dessous :

- Les clés primaires sont en **gras et soulignées**. Par simplification, nous les appelons **ID** quand l'attribut identifiant peut être généré par un numéro s'incrémentant automatiquement dans la base de données.
- Les clés étrangères sont précédées par #. Le nom de cette clé étrangère sera préfixé par ID suivi du nom de la relation dont la clé primaire est référencée par cette clé étrangère.

Ci-dessous, la liste encadrée des relations avec des remarques explicatives quand c'est jugé nécessaire. Seuls sont fournis les noms des relations et leurs attributs, sans précision des types de ces attributs que vous devrez choisir avec bon sens lors de la création de votre base de données. L'attribut NOM dans chaque relation indique un intitulé associé à l'identifiant. Quand vous voyez ... dans une relation, cela signifie que vous pouvez ajouter si besoin des attributs dans la relation.

UTILISATEUR (**ID**, EMAIL, PASSWD, NOM, PRENOM, DROIT)

Remarques : cette relation indique le droit de consultation et de mise à jour aux données de l'emploi du temps. L'attribut DROIT indique le type de droit d'accès ou/et de mise à jour aux données, comme par exemple :

- 1 pour un droit d'administrateur (service chargé de la planification) : il a le droit de consulter et de mettre à jour toutes les données de l'emploi du temps
- 2 pour un droit de référent pédagogique : il peut consulter les emplois du temps de tous les enseignants, promotions, groupes, étudiants, cours et salles MAIS ne rien mettre à jour
- 3 pour un droit d'enseignant : il ne peut consulter que son emploi du temps et le récapitulatif de ses cours sur une période
- 4 pour un droit d'étudiant : il ne peut consulter que son emploi du temps et le récapitulatif de ses cours sur une période

PROMOTION (**ID**, NOM)

GROUPE (**ID**, NOM, #ID_PROMOTION) Remarque : le groupe est affecté à une promotion

ETUDIANT (#**ID_UTILISATEUR**, NUMERO, #ID_GROUPE ...)

Remarques : l'étudiant est un utilisateur (sa clé primaire est aussi une clé étrangère qui référence la clé primaire de la relation UTILISATEUR) et est affecté à un groupe

COURS (**ID**, NOM)

TYPE_COURS (**ID**, NOM)

Remarque : cette relation indique le type de cours : par exemple cours interactif, magistral, TD, TP, projet, soutien etc.

ENSEIGNANT (#**ID_UTILISATEUR**, #**ID_COURS**) Remarques : l'enseignant est un utilisateur et peut donner plusieurs cours

SITE (**ID**, NOM) Remarque : l'école peut avoir plusieurs sites (exemple dans le cas de l'ECE : Eiffel1, Eiffel2 etc.)

SALLE (**ID**, NOM, CAPACITE, #ID_SITE) Remarque : la salle indique sa capacité et le site, si l'école dispose de plusieurs sites

SEANCE (**ID**, SEMAINE, DATE, HEURE_DEBUT, HEURE_FIN, ETAT, #ID_COURS, #ID_TYPE)

Remarques : l'attribut SEMAINE indique le numéro de semaine dans une année civile. L'attribut ETAT indique si la séance est en cours de validation, validée ou annulée : cet attribut pourra être représenté par un numéro identifiant l'état de la séance.

SEANCE_GROUPE (#**ID_SEANCE**, #**ID_GROUPE**) Remarque : on peut affecter plusieurs groupes à une séance

SEANCE_ENSEIGNANTS (#**ID_SEANCE**, #**ID_ENSEIGNANT**) Remarque : on peut affecter plusieurs enseignants à une séance

SEANCE_SALLES (#**ID_SEANCE**, #**ID_SALLE**) Remarque : on peut affecter plusieurs salles à une séance



« Gestion de l'emploi du temps d'une école »

- Idem pour une salle donnée, où on consultera tous les créneaux occupés et annulés sur une semaine pour cette salle, en indiquant toutes les informations nécessaires par créneau de séance

Emploi du temps

Salles libres

en grille

- Saisie du nom

Salle P305 - E2 (35)

Q

| 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|-------------|----|----|--|----|----|--|----|----|---|----|----|--|----|----|--|----|----|--|----|----|------|---|---|------|---|---|-----|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|
| août | | | sept. | | | oct. | | | nov. | | | déc. | | | janv. | | | févr. | | | mars | | | avr. | | | mai | | | | | | | | | | | | | | | |
| 08h30 | | | 10h00 10h15 | | | 11h45 12h00 | | | 13h30 13h45 | | | 14h15 | | | 15h15 15h30 15h45 16h00 | | | 17h00 17h15 17h30 17h45 | | | 18h | | | | | | | | | | | | | | | | | | | | | |
| lun. 01 ... | | | ANNULÉ | | | PrepAc - Anglais Mme ASSAEL ALEXANDER PrépAc Gr01B | | | Anglais 2 (app) M. BURK Ing4 apprenti SI | | | English 6 M. MAHONEY Ing3 Gr10B | | | English 4 Mme LEONARD Ing2 Gr02B | | | English 2 M. HERZ Ing1 Gr05A | | | | | | | | | | | | | | | | | | | | | | | | |
| mar. 02 | | | English 2 M. BURK Ing1 Gr04B | | | Gestion financière M. NGONGA 204-SF2 | | | REUNION Mme BLONDEL ECE PARIS - Pédagogie 2 | | | English 2 Mme LEVINE Ing1 Gr10A | | | English 2 Mme LEVINE Ing1 Gr07A | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| mer. 03 | | | English 4 Mme CLARKE Ing2 Gr01B | | | English 2 M. BYRNE Ing1 Gr08B | | | English 2 Mme LEVINE Ing1 Gr12B | | | English 4 M. REESE Ing2 Gr09A | | | English 2 Mme LEVINE Ing1 Gr11A | | | English 2 Mme LEVINE Ing1 Gr06A | | | | | | | | | | | | | | | | | | | | | | | | |
| jeu. 04 ... | | | Espagnol S2 (0) Mme ZULOAGA Espagnol grp CGN (2) Espagnol grp CGN (6) Espagnol grp CGN (8) | | | FLE - S8 M. LEROUVILLOIS FLE S8 découverte | | | FLE - S8 M. LEROUVILLOIS FLE S8 découverte | | | JPCA Mme GUYADER ECE PARIS - Admission | | | Associations ECE PARIS - BDE Mariuca de Hillerin | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ven. 05 | | | | | | | | | English 6 Mme LEONARD Ing3 Gr04B | | | English 6 M. REESE Ing3 Gr08B | | | English 6 Mme LEONARD Ing3 Gr07B | | | English 6 Mme LEONARD Ing3 Gr09A | | | | | | | | | | | | | | | | | | | | | | | | |

- Récapitulatif des cours d'un enseignant sur une période donnée (date de début et de fin) : consulter ses cours par groupe(s) (rappel : une séance de cours peut concerner plusieurs groupes), en précisant la date et le créneau horaire de la première et dernière séance sur cette période. Puis pour chaque cours et groupe(s), calculer le nombre total d'heures et de séances par cours et groupe(s) sur la période. Eventuellement, possibilité de consulter le détail de chaque séance de l'enseignant par cours et groupe(s) sur cette période : date, heure début et fin, salle, capacité, site pour chaque séance. De même, on peut calculer son volume horaire total sur cette période. Idem pour un étudiant.

Récapitulatif des cours

Saisie du nom

SEGADO JEAN-PIERRE

Q

entre le lun. 30 juil. 2018 et le sam. 03 août 2019

| Matière - Public | Première séance | Dernière séance | Durée | Nb. |
|---|-------------------------------------|-------------------------------------|-------|-----|
| Algo & prog structurée (har) - ECE PARIS IN | lun. 15 oct. 2018 de 15h30 à 17h00 | lun. 15 oct. 2018 de 15h30 à 17h00 | 01h30 | |
| Algo & prog structurée (har) - ECE PARIS IN | mer. 01 août 2018 de 08h30 à 10h00 | ven. 16 nov. 2018 de 13h45 à 15h15 | 34h30 | 23 |
| Algo & prog structurée (har) - ECE PARIS IN | lun. 30 juil. 2018 de 15h30 à 17h00 | ven. 16 nov. 2018 de 10h15 à 11h45 | 09h00 | 6 |
| Amphi d'accueil 1 - ECE PARIS ING - 1A : Ing | jeu. 13 sept. 2018 de 13h00 à 14h30 | jeu. 13 sept. 2018 de 13h00 à 14h30 | 01h30 | 1 |
| Amphi d'accueil 2 - ECE PARIS ING - 2A : In | mer. 05 sept. 2018 de 13h00 à 15h00 | mer. 05 sept. 2018 de 13h00 à 15h00 | 02h00 | 1 |
| Amphi d'accueil 2 - ECE PARIS ING - 2A : In | lun. 03 sept. 2018 de 13h45 à 15h15 | lun. 03 sept. 2018 de 13h45 à 15h15 | 01h30 | 1 |
| Amphi d'accueil 2 - ECE PARIS ING - 2A : In | mar. 21 août 2018 de 13h30 à 14h30 | mar. 21 août 2018 de 13h30 à 14h30 | 01h00 | 1 |
| Amphi d'accueil 3 - ECE PARIS ING - 3A : In | ven. 14 sept. 2018 de 14h00 à 15h30 | ven. 14 sept. 2018 de 14h00 à 15h30 | 01h30 | 1 |
| Amphi d'information | sam. 10 nov. 2018 de 09h00 à 12h00 | sam. 24 nov. 2018 de 13h45 à 17h00 | 12h30 | 4 |
| Bases de données - ECE PARIS ING - 2A : In | mer. 15 mai 2019 de 13h45 à 15h15 | mer. 05 juin 2019 de 12h00 à 13h30 | 12h00 | 8 |
| CONFERENCE | jeu. 15 nov. 2018 de 08h30 à 10h00 | jeu. 15 nov. 2018 de 08h30 à 10h00 | 01h30 | 1 |
| Informatique 1 - ECE PARIS ING - 1A : Ing1 Gr | mer. 01 août 2018 de 10h15 à 11h45 | jeu. 02 août 2018 de 12h00 à 13h30 | 03h00 | 2 |
| <div> <div>• mercredi 01 août 2018 de 10h15 à 11h45 (01h30)</div> <div>• jeudi 02 août 2018 de 12h00 à 13h30 (01h30)</div> </div> | | | | |
| Informatique 1 - ECE PARIS ING - 1A : Ing1 Gr | jeu. 20 sept. 2018 de 10h15 à 11h45 | mer. 19 déc. 2018 de 12h00 à 13h30 | 19h30 | 13 |
| Informatique 1 - ECE PARIS ING - 1A : Ing1 Gr | mer. 21 nov. 2018 de 15h30 à 17h00 | mer. 21 nov. 2018 de 17h15 à 18h45 | 03h00 | 2 |
| Informatique 1 - ECE PARIS ING - 1A : Ing1 Gr | mer. 19 sept. 2018 de 10h15 à 11h45 | mer. 19 déc. 2018 de 10h15 à 11h45 | 19h30 | 13 |
| Informatique 3 - ECE PARIS ING - 2A : Ing2 | jeu. 06 sept. 2018 de 08h30 à 10h00 | mar. 23 oct. 2018 de 15h30 à 17h00 | 10h30 | 7 |
| POO Java - ECE PARIS ING - 3A : Ing3 Gr01 | jeu. 04 avr. 2019 de 10h15 à 11h45 | lun. 03 juin 2019 de 15h30 à 17h00 | 24h00 | 16 |
| POO Java - ECE PARIS ING - 3A : Ing3 Gr01 - | ven. 10 mai 2019 de 10h15 à 11h45 | ven. 07 juin 2019 de 12h00 à 13h30 | 07h30 | 5 |
| POO Java - ECE PARIS ING - 3A : Ing3 Gr01 - | ven. 05 avr. 2019 de 08h30 à 10h00 | ven. 12 avr. 2019 de 10h15 à 11h45 | 04h30 | 3 |
| POO Java - ECE PARIS ING - 3A : Ing3 Gr05 - | lun. 06 mai 2019 de 08h30 à 10h00 | mer. 05 juin 2019 de 08h30 à 10h00 | 07h30 | |

Le module Mise à jour des données

Ce module permet de mettre à jour (modifier, insérer, supprimer) les données de la BDD via l'interface graphique de votre système : voir plus bas dans la section [Le module Interface graphique](#). Ci-dessous, quelques exemples non exhaustifs de mise à jour :

- Affecter un enseignant dans une séance de cours à condition que l'enseignant soit disponible sur le créneau de la séance. Par exemple, pour une séance sans enseignant ou en remplacement d'un autre enseignant.
- Affecter un groupe dans une séance de cours à condition que le groupe soit disponible sur le créneau de la séance et que la (les) salle(s) soient de capacité suffisante. Par exemple, pour une séance sans groupe ou en remplacement d'un autre groupe. *Remarque : l'effectif de chaque groupe peut se comptabiliser à partir des étudiants affectés au groupe.*
- Modifier le cours (son nom ou son type) dans une séance de cours
- Affecter une salle à une séance de cours à condition qu'elle soit disponible sur le créneau de la séance et que sa capacité soit suffisante pour le(s) groupe(s)
- Déplacer une séance de cours vers un autre créneau, sauf le samedi et dimanche et aux horaires d'ouverture de l'école, dans une salle libre durant ce créneau et de capacité suffisante pour le(s) groupe(s), en fonction de la disponibilité de l'enseignant et du (des) groupe(s) sur ce créneau
- Ajouter une séance de cours en lui affectant, si possible, toutes les informations nécessaires : la date (sauf samedi et dimanche), l'heure de début et de fin (en respect des créneaux horaires d'ouverture de l'école), le(s) groupe(s) et le(s) enseignant(s) disponible(s) à ce créneau horaire, la (les) salle(s) disponible(s) dont la capacité est suffisante, l'état (« en cours de validation » ou « validé »), le cours et le type de cours. *Remarque : pour la même séance, il ne peut pas y avoir de doublon de groupe, d'enseignant et de salle.*
- Ajouter un enseignant à une séance de cours à condition que l'enseignant ne soit pas déjà affecté à cette séance et qu'il soit disponible sur le créneau de la séance
- Ajouter un groupe à une séance de cours à condition que le groupe ne soit pas déjà affecté à cette séance, qu'il soit disponible sur le créneau de la séance et que la salle soit de capacité suffisante pour le(s) groupe(s)
- Annuler une séance de cours (changer l'état de cette séance à « annulée »)
- Valider une séance (changer l'état de cette séance à « validée »)
- Enlever un groupe ou/et un enseignant à une séance de cours

Le module d'accès aux données (pattern DAO)

Ce module représente la couche d'accès aux données (**DAO**) dans la BDD. Via un accès JDBC à la base de données, ce module exécute les requêtes chargés de récupérer ou de mettre à jour les données dans la base. *Il s'agit qu'un type d'objet se charge de récupérer les données dans la base et qu'un autre type d'objet soit utilisé pour manipuler ces données (couche métier).* En clair, chaque classe Java est associée à chaque table : voir [Liez vos tables avec des objets Java : le pattern DAO](#) (openclassrooms)

Le module de Reporting avec JFreeChart

Ce module permet de générer des statistiques (camemberts, histogrammes...) réactualisées des données à l'aide de la librairie **JFreeChart** : voir plus bas dans la section [Sources et ressources](#). Ci-dessous, des exemples non exhaustifs de reporting :

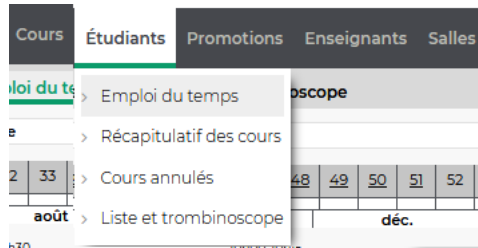
- Nombre d'heures ou de séances par cours d'une promotion sur un semestre ou une année scolaire. Idem pour un groupe
- Capacité des salles pour un site
- Taux d'occupation des salles sur une période donnée
- Etc.



Le module Interface graphique

Une fenêtre d'accueil permettra à l'utilisateur de se connecter à la BDD en saisissant son EMAIL et son PASSWD. Ces informations, si stockées dans la table UTILISATEUR, lui donneront des droits d'accès ou/et de mise à jour à certaines données de l'emploi du temps.

Votre interface graphique affichera de manière ergonomique, claire et fluide toutes les informations pertinentes que vous fournira la BDD. Elle permettra de naviguer intuitivement d'une page à une autre. Par exemple, une page de votre interface graphique pourra contenir des menus avec des items par menu, ou des onglets si vous préférez. La figure ci-dessous montre un exemple de menus (cours, étudiants etc.) et d'items d'un menu par type de recherche d'emploi du temps souhaité.



En fonction des droits de l'utilisateur, certains menus ou/et items de menu (ou onglets) seront invisibles. Par exemple, un étudiant ne pourra voir que son emploi du temps et le récapitulatif de ses cours.

Pour les mises à jour des données, uniquement possibles pour un administrateur, il faut tout d'abord sélectionner votre menu et item de menu (ou onglet) qui va ouvrir une page de mise à jour. Avant la mise à jour, la page devra d'abord rechercher les informations souhaitées. Par exemple, si on souhaite affecter un enseignant à une séance, il faut d'abord chercher cette séance et cet enseignant. Puis si c'est possible, ajouter cet enseignant à cette séance, par exemple avec un bouton de validation.

L'affichage de l'emploi du temps pourra se représenter sous différentes formes :

- Forme classique en grille : par exemple une case par jour et heure, le créneau horaire d'une séance de cours pouvant être à cheval sur 2 cases. Pour chaque séance, on indiquera le cours, le(s) groupe(s), la (les) salle(s), la capacité, le site, le(s) enseignant(s). Possibilité d'afficher la grille avec les jours en lignes et les heures en colonnes ou vice-versa.

[illegible]

« Gestion de l'emploi du temps d'une école »

- Liste : les jours se suivent en lignes et sous chaque jour, afficher les séances de cours en lignes. Pour chaque séance, on indiquera le créneau horaire, le cours, le(s) groupe(s), la (les) salle(s), la capacité, le site, le(s) enseignant(s).

Emploi du temps

Récapitulatif des cours

en liste

- Saisie du nom

SEGADO JEAN-PIERRE

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------|----|----|----|-------|----|----|----|------|----|----|----|------|----|----|----|------|----|----|----|-------|----|---|---|-------|---|---|---|------|---|---|----|----|----|----|----|
| 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
| août | | | | sept. | | | | oct. | | | | nov. | | | | déc. | | | | janv. | | | | févr. | | | | mars | | | | | | | |

mardi 02 avril 2019

a

08h30 - 10h00 POO Java

Ing3 Gr05
Ing3 Gr06
Ing3 Gr07

CNAM - Salle 30-1.04

COURS

jeudi 04 avril 2019

j

10h15 - 11h45 POO Java

Ing3 Gr01

Labo P416 - E2 (I)

N°1 COURS

k

12h00 - 13h30 Théorie des graphes

Ing2 Gr03

Salle P348 - E2 (40)

N°1 COURS

l

15h30 - 17h00 PrepAc - Informatique

Prép/Ac Gr01

Labo P314 - E2 (A)

N°1 COURS

vendredi 05 avril 2019

m

08h30 - 10h00 POO Java

Ing3 Gr01
Ing3 Gr02
Ing3 Gr03
Ing3 Gr04
Ing3 Gr05

Salle EM009 Amphi - EI
Salle EM010 Amphi - EI

N°1 COURS

m

10h15 - 11h45 POO Java

Ing3 Gr06
Ing3 Gr07
Ing3 Gr08
Ing3 Gr09
Ing3 Gr10

Salle EM009 Amphi - EI
Salle EM010 Amphi - EI

N°1 COURS

l

15h30 - 17h00 POO Java

Ing3 Gr01

Labo P314 - E2 (A)

N°2 COURS

k

17h15 - 18h45 Théorie des graphes

Ing2 Gr03

Salle P348 - E2 (40)

N°2 COURS

Contraintes techniques

L'implémentation de la gestion informatique de l'emploi du temps va être réalisée en **Java (JDK 1.8)** en utilisant le **versioning GIT**. L'exécutable **.jar** de votre programme sera en mode graphique. Votre code devra être commenté au format **javadoc**, surtout devant chaque classe et méthode. La base de données (BDD) sera stockée sur **WampServer** : voir dans la section [Création de la BDD](#).

Sources et ressources

Supports de cours et du projet sur campus : [POO Java](#) et section [Ressources](#) de la page [Projet POO Java](#)

Architecture de conception technique :

[Modèle Vue Contrôleur](#) (wikipedia)

Adopter une architecture MVC (openclassrooms)

Liez vos tables avec des objets Java : le pattern DAO (openclassrooms)

JDBC : <https://www.jmdoudoux.fr/java/dej/chap-jdbc.htm>, (Auteur: Jean-Michel Doudoux)

JDBC2020.zip zip du code de JDBC (Java Data Base Connectivity) vu en amphi

My SQL : <http://dev.mysql.com/doc/refman/5.7/en/>

Wampserver : Tutoriel WampServer

Git : [Slides - Versioning avec Git](#) et [TP - Introduction à Git](#) vus en formation de versioning

JFreeChart :

[The JFreeChart Class Library](#) (Auteur: David Gilbert)

<http://www.jfree.org/jfreechart/api/javadoc/index.html>

<http://www.java2s.com/Code/Java/Chart/CatalogChart.htm>

<http://www.jfree.org/forum/>

Wireframe: <https://webdesign.tutsplus.com/articles/a-beginners-guide-to-wireframing--webdesign-7399>

Storyboard: <https://www.fastcodesign.com/1672917/the-8-steps-to-creating-a-great-storyboard>



Equipes et planning

Règles de constitution des équipes :

Les membres d'une équipe peuvent appartenir à différents groupes de TD, avec 2 ou 3 étudiants par équipe.

Planning :

Ce projet s'étale de la semaine du 11 mai à la semaine du 1er juin 2020. Le dépôt des livrables est fixé au 7 juin au plus tard : voir section Livrables à déposer sur campus : PowerPoint et code. Les soutenances auront lieu dans la semaine du 8 juin. Des créneaux de passage vous seront fournis ultérieurement.

Travail à faire et planning

Phase de conception (patterns MVC, DAO et maquettes graphiques) AVANT la phase de réalisation

- 1) En utilisant l'architecture des patterns **MVC** et **DAO**, élaborer le diagramme de classes, avec d'abord les relations adéquats (simple association, référence, agrégation, composition ou héritage) et les attributs (pas d'attribut objet !), sur lesquels sont basées les données dans votre système de l'emploi du temps.
- 2) Définir les classes qui interviennent dans chacun des 3 types de modules **MVC**.
- 3) Définir les fonctionnalités (méthodes) nécessaires pour chaque classe de vos 3 types de modules **MVC** de votre système, en spécifiant pour chacune les paramètres en entrées et sorties. En respect du pattern **DAO**, définir des classes qui sont liées aux données de la BDD à l'aide des méthodes de récupération, d'insertion, de modification et de suppression.
- 4) Illustrer les maquettes de votre interface graphique c'est-à-dire le *storyboard* et les *wireframes* : voir « Powerpoint avec les slides suivants » dans la section suivante.

Livrables à déposer sur campus : PowerPoint et code

1) Powerpoint avec les slides suivants :

- Page de garde avec titre, noms coéquipiers et groupe de TD (1 slide)
- Sommaire (1 slide)
- **Répartition des tâches** par fonctionnalités sous forme de tableau (1 slide)
- **Diagramme de classes** avec l'outil [Draw.io](https://draw.io) ou équivalent, selon les patterns **MVC** et **DAO** et présentant les attributs (pas d'attribut objet !), les méthodes, les cardinalités, sans constructeurs ni getters/setters (1 slide ou plus si le diagramme est illisible : par exemple, 1 slide présente l'architecture générale du pattern MVC avec seulement les noms des classes, puis un slide pour chacun des 3 types de modules MVC détaillant le contenu des classes).
- **Design de la maquette de votre interface graphique** principalement composée des 2 éléments suivants :
 - Le *storyboard* : liens entre les pages, symbolisés par une flèche pour naviguer d'une page à une autre (1 à 2 slides)
 - Des *wireframes* de certaines de vos pages (par exemple, 1 page pour une recherche, 1 pour une mise à jour et 1 autre pour le Reporting) : composants graphiques Swing légendés avec les conteneurs encadrés, leur mise en page *layout* (au plus 3 slides)
- **Versioning GIT** : screenshot et lien avec login et passwd, montrant clairement la bonne utilisation et la répartition des tâches sur la (ou les) version(s) du code partagé entre coéquipiers (1 slide)
- **Bilan individuel et collectif** sans blabla (par exemple sous forme de tableau) sur l'état du travail effectué, des compétences acquises et des points d'amélioration. (1 à 2 slides).
- **Sources** : web avec les liens, livres, supports de cours en citant les auteurs. **Toute source non citée est considérée de facto comme un plagiat.** (1 slide)



2) Code :

- Tous les dossiers et fichiers du projet développé sur Eclipse ou Netbeans avec vos sources **.java**, les librairies **.jar** nécessaires (exemples : accès au serveur de la BDD, JFreeChart, etc.), l'exécutable **.jar** en mode graphique de votre programme, la documentation **Javadoc** commentée pour les classes et les méthodes (respectez bien le format Javadoc des commentaires au-dessus de chaque classe et méthode de votre code) et tout les fichiers utiles au bon fonctionnement de votre projet (exemples : images, etc.)
- Le **script du fichier des requêtes SQL** exporté à partir de votre base de données sur WampServer

Le PowerPoint et le code sont à déposer sur campus le **dimanche 7 juin 2020 à 23h55 dernier délai** dans la section [Livrable à déposer](#) de la page campus [Projet POO Java](#). Ils seront archivés dans un fichier **.zip** ou **.rar** portant les noms des coéquipiers, avec un seul dépôt par équipe. **Tout code sans les fichiers sources .java vaudra 0. 2 pts de pénalité sur la note du projet par jour de retard. Tout plagiat sera sévèrement sanctionné par 0 et un avertissement.**

Déroulement de la soutenance

La soutenance du projet Java devra respecter les consignes suivantes :

- La durée maximale de présentation de la démo et du PowerPoint sera en tout de 20 minutes maximum par équipe, questions du jury comprises.
- Vos jurys auront reçu les grilles d'évaluation de la démo et du PowerPoint.
- L'ordre de passage vous sera attribué pour chaque séance de soutenances de 1h30. **Tous les étudiants de la séance doivent être présents sous peine d'être pénalisés par 0.**
- En cas d'absence justifiée d'étudiants, avec justificatif valable, l'équipe pourra présenter la soutenance hors séances de soutenances à la convenance du jury et de l'équipe.
- Le temps de parole doit être réparti équitablement entre chaque membre de l'équipe.
- Les étudiants qui auront été affectés d'office, à cause d'un retard d'inscription aux équipes, seront très particulièrement contrôlés lors de leur soutenance du projet Java. Ce qui ne veut pas dire que les autres étudiants ne seront pas aléatoirement ou intuitivement contrôlés ☺
- Télécharger de campus son livrable, sauf si le chargé de TP aura eu la bonté de le faire avant ☺ Puis dézippez le.
- Démarrer votre présentation PowerPoint. Durée max : 5 minutes, questions du jury comprises.
- Démarrer votre démo en double-cliquant sur le **jar** pour lancer la fenêtre d'accueil graphique de connexion à votre BDD, avec saisie de l'EMAIL et du PASSWD.
- La durée de la démo est de 15 minutes maximum, questions du jury comprises. Elle doit surtout montrer les 10 points suivants :

- 1) La qualité de la présentation orale (clarté dans les explications, cohérence de l'argumentation, cohésion de l'équipe) avec un petit scénario de fonctionnement de votre système préparé à l'avance, en tenant compte des critères suivants et de la durée maximale de présentation.
- 2) La fiabilité de votre système et la pertinence des informations affichées
- 3) La qualité visuelle, la fluidité, l'ergonomie et l'originalité particulière de l'interface graphique.
- 4) La bonne interaction de votre programme Java avec la Base de Données.
- 5) Une recherche sur une table de votre choix (exemples au choix : ETUDIANT, COURS, SITE) avec filtrage des informations autres que le numéro identifiant (exemple : NOM ou mieux à cause des homonymes, parmi une liste).
- 6) Une recherche croisée sur plusieurs tables (exemples évidents au choix : l'emploi du temps d'un ETUDIANT, ENSEIGNANT, PROMOTION, GROUPE ou SALLE) avec filtrage des informations autres que le numéro identifiant.



- 7) Une mise à jour (modification, insertion et suppression) de un à plusieurs enregistrements (lignes) d'une table de votre choix. Pour preuve de la fiabilité de cette MAJ, faire la recherche sur cette table avec filtrage des informations des données mises à jour. Cette MAJ doit également respecter les contraintes d'intégrité référentielle (foreign key).
- 8) Les différents droits d'accès ou/et de mise à jour aux données selon l'utilisateur qui se connecte.
- 9) Démo du module de Reporting avec **JFreeChart** : ce module doit montrer sous formes de graphes (camemberts, histogrammes etc.) des statistiques prédéfinies par votre interface
- 10) A la fin de la démo, vous présenterez brièvement votre **Javadoc** commentée.

Attention, cette soutenance vise à vérifier si votre travail répond aux exigences du client. Ce dernier (le jury) **teste votre projet sur la version déposée sur campus dans une machine de l'école** et en vérifie l'adéquation avec ses attentes. S'il « achète » votre projet, vous avez une bonne note, sinon... elle sera mauvaise. Si vous avez développé sur un système particulier, venez avec votre matériel. Un plantage le jour du test client est du plus mauvais effet alors préparez-vous un peu à l'avance et non à la dernière minute.

Critères d'évaluation du travail à présenter

En respect des consignes spécifiées précédemment, chaque équipe sera évaluée sur les critères suivants :

- La **démo** de votre code **en adéquation avec celui déposé sur campus**. Elle est **notée sur 20 avec un coefficient de 2**. Vous serez jugés sur les points encadrés à la page précédente dans la section [Déroulement de la soutenance](#) et la **Javadoc** (montrant la structure modulaire et les commentaires de votre code).
- Le **PowerPoint** sera **noté sur 20 avec un coefficient de 1**. Vous serez jugés sur le respect du plan des slides donné dans la section [Livrables à déposer sur campus : PowerPoint et code](#). Mais aussi, comme la démo, sur la qualité de la présentation orale.
- Des **pénalités** pourraient être appliquées sur la note de la soutenance dans les cas suivants : retard de dépôt du livrable (PowerPoint et code) avec **-2 points par jour de retard**, fichiers manquants dans le code déposé, plagiat, absence à une soutenance, prestation individuelle peu ou pas convaincante.
- La **note finale sera égale à (démo * 2 + PowerPoint) / 3** sans tenir compte des éventuelles pénalités

Recommandations

Le jour de la soutenance, vous devrez faire tourner le système que vous aurez déposé sur campus. **Aucune amélioration ou nouvelle version développée ultérieurement ne sera acceptée.**

Pensez, après le dépôt sur campus, à télécharger votre dépôt sur une machine de l'école pour le faire tourner dans les conditions de la soutenance et pallier les anomalies.

Ne vous laissez pas abuser par l'illusion d'avoir beaucoup de temps. Respectez l'organisation prévue pour avoir terminé avant les vacances. Commencez suffisamment tôt pour vous organiser au mieux et faire quelque chose de bien. Ne soyez pas trop ambitieux au départ. Préférez faire proprement un projet simple et bien testé avant de partir dans les extensions. Travaillez par version en utilisant un logiciel de « versioning » et ne faites des modifications que sur une nouvelle version en conservant précieusement la dernière qui marche. C'est toujours trop bête de planter le jour de la soutenance.

Ne sous-estimez pas les phases de test qui doivent être régulières et sont presque aussi importantes que les phases de développement.

Pensez à faire appel à vos amis ou votre famille pour tester et critiquer ce que, vous-même, ne pouvez critiquer objectivement.

Bon projet à tous

