

BREVET DE TECHNICIEN SUPÉRIEUR
SERVICES INFORMATIQUES AUX ORGANISATIONS
Option : Solutions logicielles et applications métiers

**U5 – PRODUCTION ET FOURNITURES DE
SERVICES INFORMATIQUES**

SESSION 2021

Durée : 4 heures
Coefficient : 5

Matériel autorisé :

Aucun matériel ni document est autorisé.

Dès que le sujet vous est remis, assurez-vous qu'il est complet.

Le sujet comporte 18 pages, numérotées de 1/18 à 18/18
(sans compter la page de garde).

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 0 sur 18

Hackat'Innov

Le sujet compte 18 pages dont 10 pages de documentation.

Le sujet est constitué de quatre dossiers qui peuvent être traités de façon indépendante.

Présentation du sujet	2-8
Dossier documentaire	9-18

Dossier documentaire

DOCUMENT 1 : base de données actuelle de l'application <i>Hackat'Orga</i>	9
DOCUMENT 2 : extrait de la syntaxe <i>SQL</i>	10
DOCUMENT 3 : exemple d'événements au format <i>JSON</i>	10
DOCUMENT 4 : diagramme de classes du module <i>Hackat'Event</i>	11
DOCUMENT 5 : code <i>PHP</i> partiel des classes du module <i>Hackat'Event</i>	12
DOCUMENT 6 : extrait de la classe de test <i>TestInitiation</i>	15
DOCUMENT 7 : extrait de documentation <i>PHP</i>	15
DOCUMENT 8 : schéma relationnel de la base de données <i>Hackat'Event</i>	16
DOCUMENT 9 : interface graphique affichant la liste des hackathons	17
DOCUMENT 10 : fichier <i>liste.vue</i>	18
DOCUMENT 11 : interface de programmation (<i>API</i>) <i>api-rechercher.php</i>	18

Barème

Dossier A	Gestion des participants	30 points
Dossier B	Gestion d'évènements organisés autour des hackathons	35 points
Dossier C	Gestion des recherches pour le mobile	20 points
Dossier D	Gestion des votes	15 points
	TOTAL	100 points

Présentation du contexte

Contraction de « hack » et de « marathon », l'hackathon est un processus créatif utilisé dans le domaine de l'innovation numérique. Durant généralement un week-end, des développeurs volontaires se réunissent pour faire de la programmation informatique en mode collaboratif sur un thème défini. L'engouement pour ce type de manifestation témoigne d'une volonté de développer toujours davantage le numérique au service de la société.

Nous distinguerons les phases du déroulement d'un hackathon, sur le week-end, des étapes de son organisation, qui commence bien en amont.

Les phases du déroulement d'un hackathon

Même s'ils peuvent porter sur des thèmes variés, les hackathons se déroulent toujours selon un même schéma :

- l'accueil des participants et la présentation (thème, projets retenus, constitution des équipes, planning, jury, etc.) constituent la première phase du hackathon ;
- les équipes travaillent ensuite sur le développement de leur projet : remue-méninges, maquettage, prototypage, etc. Lors de ces phases de travail, les équipes sont accompagnées par des mentors (*coachs*) de différents domaines (développement, graphisme, communication, etc.) ;
- à l'issue de ce travail intensif, chaque équipe présente son prototype au jury ;
- enfin, la délibération du jury puis la proclamation des résultats et la remise des récompenses constituent la dernière phase du hackathon.

Un hackathon est également souvent l'occasion d'organiser des événements satellites tels que des conférences, des ateliers d'initiation ou encore des tables rondes. Ces événements satellites ont un double objectif :

- permettre aux membres des équipes de développer ou d'acquérir des connaissances et des compétences utiles pour leurs projets ;
- attirer du public extérieur pour le sensibiliser aux thématiques abordées et faire connaître le hackathon.

Les étapes de l'organisation d'un hackathon

Le processus d'organisation, essentiel pour assurer le bon déroulement du hackathon, se décompose en plusieurs étapes :

- initialisation : choix du lieu, de la date et du thème, composition du jury, contact avec les organisateurs ;
- publication du hackathon : édition et mise en ligne du planning ;
- inscription des participants : ouverture des inscriptions en ligne ;
- choix des projets : détermination des projets retenus en fonction du positionnement des participants et constitution des équipes ;
- lancement : lancement du hackathon avec choix de la ou du chef de projet de chaque équipe, démarrage ;
- clôture : livraison des prototypes par les équipes pour la présentation au jury de chacun par sa ou son chef de projet ;
- résultats : gestion des votes des membres du jury puis édition des résultats.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 2 sur 18

L'organisation cliente et le prestataire de service informatique

Pour répondre au développement des hackathons en France, Mme Majdouline Mabilie a créé, il y a deux ans, une entreprise jeune-pousse (*start-up*) Hackat'Innov qui a pour but de simplifier la gestion de l'organisation d'hackathons. Son entreprise a reçu un prix de l'innovation numérique et a donc pu lever des fonds nécessaires à l'embauche de deux personnes supplémentaires.

Le service informatique de l'entreprise, composé de trois personnes, est chargé de réaliser les projets d'évolution du système d'information d'Hackat'Innov.

Les projets d'évolution du système d'information

Un des principaux projets de Hackat'Innov est le développement d'une application *Web Hackat'Orga* qui permet de gérer les étapes du processus d'organisation d'un hackathon : initialisation, publication, inscription, choix des projets, lancement, clôture et résultats. La première étape (l'initialisation d'un hackathon) a été développée dans l'application lors d'une première itération. Aujourd'hui, Mme Mabilie souhaite :

- finaliser l'application *Hackat'Orga* pour gérer l'ensemble des étapes (à l'exception du choix des projets) ;
- développer un module *Hackat'Event* permettant la gestion des événements satellites organisés lors des hackathons.

Par ailleurs, à la demande des organisateurs d'hackathons et des participants, Hackat'Innov a lancé son service informatique dans le développement d'applications mobiles :

- une première application permettant de consulter les informations sur les hackathons gérés par Hackat'Innov est en cours de développement ;
- le développement d'une seconde application mobile, dédiée à la gestion des votes, est à l'étude.

En tant que stagiaire chez Hackat'Innov, vous participez à ces quatre évolutions sous la responsabilité de Mme Majdouline Mabilie.

Vous vous appuyerez sur le dossier documentaire mis à votre disposition.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 3 sur 18

Dossier A – Gestion des participants

Documents à utiliser : 1, 2

L'application *Web Hackat'Orga* permet actuellement de gérer uniquement l'étape d'initialisation d'un hackathon. Mme Mabile vous demande de la faire évoluer pour y intégrer les étapes de publication et d'inscription des participants.

Mission A.1 – Évolution de la base de données pour la gestion des hackathons

IMPORTANT : la candidate ou le candidat présentera les évolutions de la structure de la base de données en adoptant le formalisme de son choix (schéma entité-association, diagramme de classes, ou encore schéma relationnel).

L'application *Hackat'Orga* utilise une base de données dont le schéma complet vous est fourni dans le dossier documentaire. Il est nécessaire de faire évoluer la structure de cette base de données pour qu'elle intègre les données utiles à l'inscription en ligne des participants et à l'édition du planning détaillé des hackathons.

Inscription des participants

Pour participer à un hackathon, il est indispensable de s'inscrire. Les organisateurs ont défini, pour chaque hackathon, un nombre de places limité et une date limite à respecter pour l'inscription des participants. Ces derniers peuvent consulter les caractéristiques du hackathon sur la partie publique de l'application *Web* et s'y inscrire via un formulaire.

Pour s'inscrire, il faut se connecter avec un profil de membre participant. Celui ou celle qui n'en possède pas en crée un, en renseignant son nom, prénom, mél, téléphone, date de naissance et un lien vers son portfolio afin de mieux se faire connaître. Ce profil est enregistré et peut être repris d'un hackathon à l'autre.

Ensuite, il faut sélectionner le hackathon auquel participer puis saisir un texte libre présentant une compétence ou un outil répondant au thème indiqué, information qui sera utile lors de la constitution des équipes.

À l'issue de l'inscription, la date de saisie est mémorisée et un numéro unique d'inscription est attribué séquentiellement au sein de chaque hackathon.

Constitution des équipes

Les inscrits se regroupent en équipes autour des projets retenus pour réaliser un prototype d'un projet. Un inscrit ne peut être affecté qu'à une seule équipe. Une équipe travaille sur un seul projet et choisit un nom d'équipe. Plusieurs équipes peuvent choisir le même projet, elles seront alors identifiées relativement au projet choisi. Une ou un chef de projet est défini dans chaque équipe parmi ses participants.

Le hackathon démarre alors et les différentes phases en présentiel peuvent avoir lieu ; ces phases ne sont pas à gérer dans cette mission.

Vote des membres du jury

À l'issue du temps imparti pour le hackathon, chaque équipe fournit un lien vers son prototype. Lors de la phase de présentation, chaque équipe fait une démonstration de son prototype au jury. Chaque membre du jury doit alors attribuer une note de 0 à 5 pour chaque équipe ; ce vote est conservé. Le classement final se base sur le nombre de points obtenus pour récompenser les meilleures équipes.

Remarque : les membres du jury ont été choisis lors de la phase d'organisation, sachant qu'un membre du jury peut être aussi membre participant mais pour d'autres hackathons.

Question A.1

Proposer une modification de la base de données utilisée par l'application existante *Hackat'Orga* prenant en compte les inscriptions, la gestion des équipes et le vote final.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 4 sur 18

Mission A.2 – Édition du planning des phases d'un hackathon

L'ordre de déroulement des différentes phases (accueil des participants, travail en autonomie, etc.) s'applique à tous les hackathons. Les participants souhaitent pouvoir accéder au planning contenant les horaires de début et de fin prévues pour chacune des phases d'un hackathon.

Exemple :

Vendredi 11 juin	...	Dimanche 13 juin
17h – 19h : Accueil des participants 19h – 24h : Travail en autonomie	...	00h – 12h : Travail en autonomie 12h – 14h : Repas festif 14h – 16h : Présentation des projets 16h – 17h : Délibération et résultats

Pour répondre à ce besoin, la base de données a été complétée de la manière suivante :

PHASE (id, libelle)

clé primaire : id.

Domaines des données utilisées :

- id : numéro séquentiel ;
- libellé : chaîne de 100 caractères maximum.

PLANNING (idHackathon, dateHeureDebut, idPhase, duree)

clé primaire : idHackathon, dateHeureDebut.

clés étrangères : idHackathon en référence à id de HACKATHON ;
idPhase en référence à id de PHASE.

Domaines des données utilisées :

- idHackathon : numéro séquentiel ;
- idPhase : nombre entier ;
- dateHeureDebut : horodatage ;
- duree : nombre entier d'heures.

Mme Mabile attire votre attention sur la règle suivante : « Il ne peut y avoir qu'une seule phase qui débute à une heure donnée ».

Question A.2.1

Justifier que la structure de la base de données permet d'implémenter cette règle de gestion.

Question A.2.2

Écrire la requête de création de la table PLANNING.

Dossier B – Gestion d'évènements organisés autour des hackathons

Documents à utiliser : 3, 4, 5, 6, 7

IMPORTANT : la candidate ou le candidat peut choisir de présenter les éléments de code à l'aide du langage de programmation de son choix ou de pseudo-code algorithmique.

Afin d'apporter de l'attractivité auprès d'un plus large public, Hackat'Innov a décidé d'organiser des évènements satellites lors des hackathons. Pour gérer ces évènements qui peuvent être des conférences comme des initiations aux outils numériques animées par une ou un intervenant, un nouveau module *Hackat'Event* doit être développé.

Mme Mabilille souhaite que vous implémentiez les classes métier de ce futur module.

Le diagramme de classes de l'application est fourni dans le dossier documentaire. Les évènements de type « initiation » seront soumis à inscription car le nombre de participants est limité pour ce type d'évènement.

On s'intéresse ici à la classe Initiation. Les initiations nécessitent que les participants apportent une liste de matériels et de logiciels prédéfinis. Cela peut-être des fournitures ainsi que certains logiciels à installer. Certaines méthodes de cette classe retournent une chaîne au format *JSON (JavaScript Object Notation)* afin de faciliter les échanges entre les différents applicatifs, en particulier les applicatifs mobiles.

La méthode de développement de ce projet est centrée sur les tests. Le développement piloté par les tests (*TDD - Test Driven Development*) conduit à écrire d'abord les tests et ensuite les méthodes associées. Pour s'assurer de la validité de chaque méthode, celle-ci devra passer tous les tests.

L'écriture des tests est achevée ; il faut maintenant écrire les méthodes qui réussissent ces tests. Vos collègues ont commencé, il faut compléter leur travail. Voici la liste des erreurs relevées lors de la dernière exécution des tests sur la classe Initiation :

- Le test *testAjouterMaterielQuantiteZero* signale une erreur dans la méthode *ajouterMateriel* de la classe Initiation, le message retourné est « Matériel ajouté alors que la quantité est égale à 0 ».
- Le constructeur de la classe Initiation est manquant.
- Le test *testAjouterParticipantLimiteNombrePlaces* ne renvoie rien de concluant car la méthode *ajouterParticipant* de la classe Initiation n'est pas écrite. La signature de cette méthode n'est d'ailleurs pas définie.
- Il manque également la méthode *lesParticipantsToJson* dans la classe Initiation.
- La méthode *toJson* de la classe Initiation n'est pas écrite non plus. Les spécifications demandent que toutes les caractéristiques d'une séance d'initiation apparaissent, ainsi que la liste des matériels et des participants sauf s'il n'y a pas de participants.

Question B.1 – Corrections des erreurs signalées ci-dessus

- Corriger la méthode *ajouterMateriel*.
- Écrire le constructeur de la classe Initiation.
- Écrire la méthode *ajouterParticipant*.
- Écrire la méthode *lesParticipantsToJson*.
- Écrire la méthode *toJson*.

Question B.2 – Gestion du planning des évènements des membres au sein de l'hackathon

Décrire la structure du dictionnaire retourné par la méthode *planningParParticipant* de la classe Hackathon.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 6 sur 18

Dossier C – Gestion des recherches pour le mobile

Documents à utiliser : 2, 4, 8, 9, 10, 11

De nombreux organisateurs, lors des phases de contact, interrogent Mme Mabilles pour obtenir des informations concernant les hackathons que gère déjà Hackat'Innov.

Elle a demandé au service informatique de produire une application mobile pour obtenir en temps réel sur un téléphone connecté à l'internet les informations. Cette application utilise l'outil de développement (*framework*) JavaScript *Vue.js* pour la production des interfaces graphiques sur mobile multifonction (*smartphone*) et la base de données issue de l'application *Hackat'Event*.

Le document 9 reprend une copie d'écran de l'interface générée qui permet de rechercher la liste des hackathons par ville (champ texte libre). La recherche est traitée par une interface de programmation (*API - Application Programming Interface*) détaillée dans le document 11.

Question C.1

Expliquer par une phrase l'objectif du code encadré qui vient d'être rajouté dans le fichier *liste.vue* (document 10).

On désire compléter l'affichage en ajoutant une colonne « thème » à droite de la colonne « ville ». Le champ « recherche » porte sur l'une ou l'autre de ces deux colonnes.

Question C.2

Compléter le code du fichier *liste.vue* (document 10) et l'interface de programmation (*API*) correspondante en indiquant les lignes concernées.

Afin de gérer au mieux les événements, Mme Mabilles envisage de compléter cette application mobile par d'autres fonctionnalités comme la recherche de membre de jury, pour un remplacement de dernière minute. Pour cela, vous êtes en charge de la programmation dans l'interface de programmation (*API*) des requêtes et/ou procédures nécessaires.

La requête *SQL* renvoyant le libellé, le nombre de places des événements de type « initiation » et le thème des conférences a déjà été écrite :

```
SELECT libelle, nbplaces, theme
FROM EVENEMENT E
LEFT OUTER JOIN INITIATION I ON E.id = I.idEvenementInit
LEFT OUTER JOIN CONFERENCE C ON E.id = C.idEvenementConf;
```

Question C.3

a. Proposer un commentaire expliquant l'utilité des jointures de type **LEFT OUTER JOIN** dans cette requête.

b. Compléter cette requête *SQL* en ajoutant la date, l'heure et la salle de l'évènement avec le nom et le prénom de l'animateur. Les événements seront triés chronologiquement.

Afin d'améliorer le choix des événements, on désire savoir s'ils connaissent du succès ou peu ou pas du tout (aucun inscrit).

Question C.4

Créer une vue en langage *SQL* permettant de retourner les identifiants de tous les événements de type « initiation » avec, pour chacun, le nombre de membres inscrits.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 7 sur 18

Dossier D : Gestion des votes

À terme, Hackat'Innov souhaite proposer en téléchargement une application mobile type *Android* :

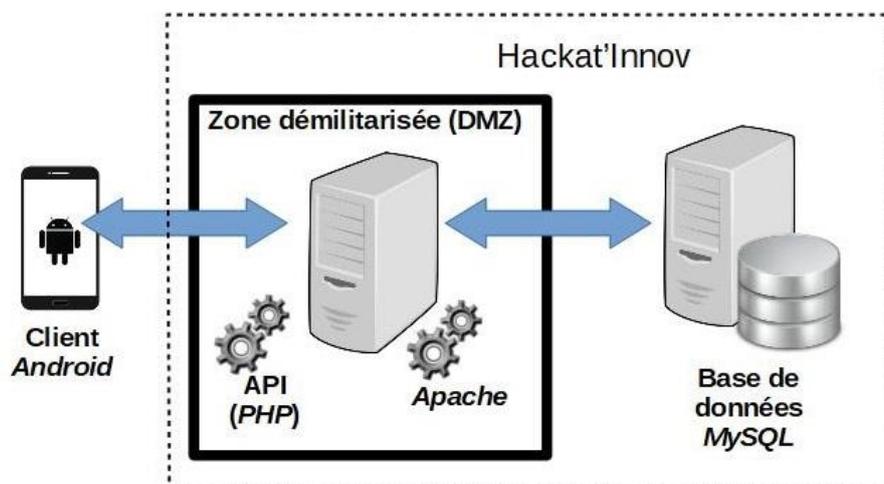
- permettant aux participants de se positionner sur les projets proposés pour un hackathon ;
- permettant aux membres du jury de voter, lors d'un hackathon, pour les trois meilleurs prototypes tous projets confondus (sachant qu'un membre de jury peut être membre participant pour d'autres hackathons que ceux pour lesquels il est membre du jury).

L'application de gestion des votes a été identifiée comme particulièrement sensible.

Question D.1

Décrire une conséquence d'utilisation malveillante de l'application de gestion des votes.

Mme Mabile a commencé une étude de l'architecture applicative et a réalisé le schéma suivant :



Question D.2

Préciser les intérêts de mettre le serveur de bases de données en dehors de la zone démilitarisée (*DMZ*) pour aider Mme Mabile à compléter son étude.

Suite à un audit sur le thème de la cybersécurité, audit appuyé sur *OWASP (Open Web Application Security Project)*, des points de vigilance ont été relevés :

1. Sécuriser le service *HTTP* (utiliser *HTTPS*).
2. Utiliser des requêtes préparées.
3. Se prémunir des attaques par injection *SQL*.
4. Se connecter via un réseau *Wi-Fi* sécurisé (pour éviter des attaques de type *Man in the Middle*).
5. Préserver l'intégrité des données (configurations et mise en place des droits d'accès).
6. S'assurer de la mise à jour des bibliothèques logicielles et des composants logiciels.

Question D.3

Présenter sous forme d'un tableau le ou les éléments de l'architecture applicative concerné(s) par chacun des points de vigilance relevés.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 8 sur 18

DOCUMENT 1 : base de données actuelle de l'application *Hackat'Orga*

Schéma conceptuel des données :

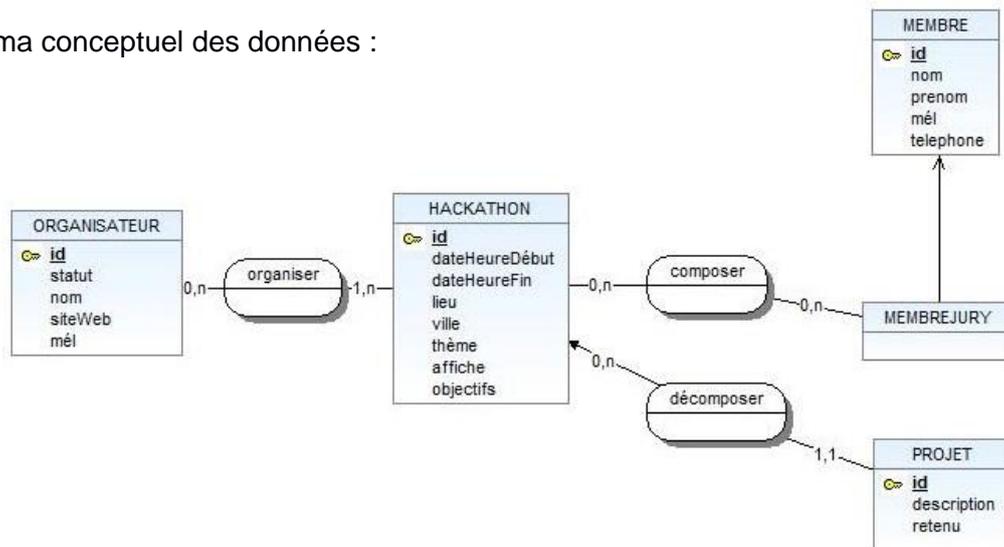


Diagramme de classes :

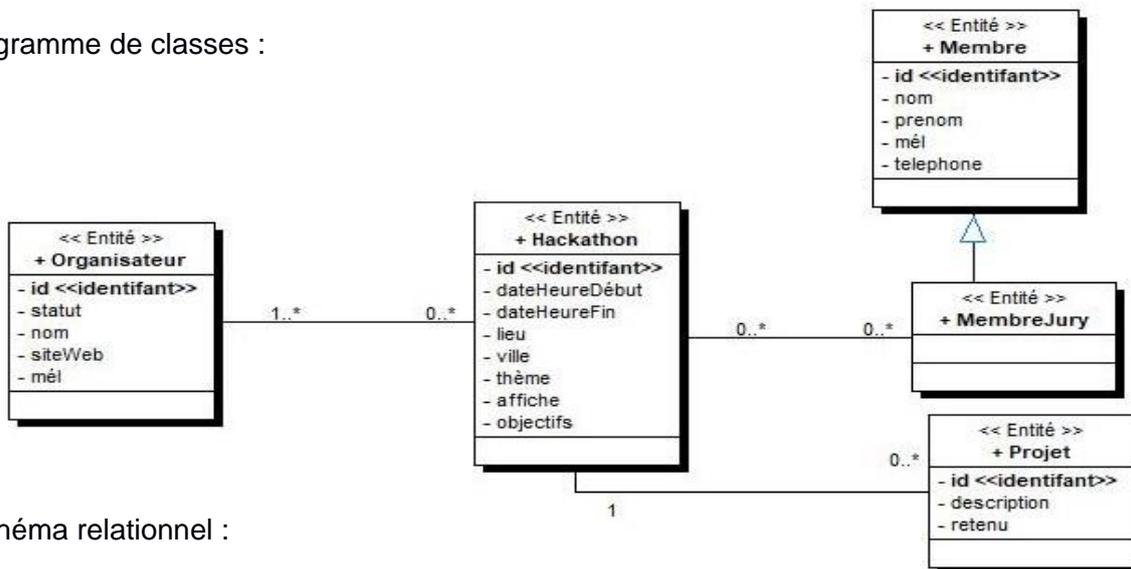


Schéma relationnel :

ORGANISATEUR (id, statut, nom, siteWeb, mel)

clé primaire : id

statut pourra prendre comme valeurs : entreprise, collectivité, association, organisme de formation, etc.

MEMBRE (id, nom, prenom, mel, telephone)

clé primaire : id

HACKATHON (id, dateHeureDebut, dateHeureFin, lieu, ville, theme, affiche, objectifs)

clé primaire : id

ORGANISER (idHackathon, idOrganisateur)

clé primaire : idHackathon, idOrganisateur

clé étrangère : idHackathon en référence à id de HACKATHON

idOrganisateur en référence à id de ORGANISATEUR

COMPOSER (idHackathon, idMembreJury)

clé primaire : idHackathon, idMembreJury

clé étrangère : idHackathon en référence à id de HACKATHON

idMembreJury en référence à id de MEMBRE

PROJET (id, description, retenu, idHackathon)

clé primaire : id

clé étrangère : idHackathon en référence à id de HACKATHON

Remarque : retenu indique si le projet proposé a été retenu ou pas.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : S15SLAM	Page 9 sur 18

DOCUMENT 2 : extrait de la syntaxe SQL

CREATE TABLE nomTable

```
(champ1 type [NOT NULL/NULL] [,champ2 type ...]
[CONSTRAINT nom_contrainte PRIMARY KEY (champ[, ...]) |
CONSTRAINT nom_contrainte FOREIGN KEY (champ [, ...]) REFERENCES nomTable (champ [, ...])
  [ON DELETE {CASCADE | RESTRICT | SET NULL}]
  [ON UPDATE {CASCADE | RESTRICT | SET NULL}] ]
);
```

CREATE VIEW nomVue (champ[, ...]) **AS** requête SQL ;

DOCUMENT 3 : exemple d'événements au format JSON

Exemple d'évènements de type initiation :

```
[{
  "libelle" : "Introduction au PHP",
  "dateHeure" : "19/06/2021 13:00",
  "duree" : "2h",
  "salle" : "Alan Turing",
  "typePublic" : "étudiants ou jeunes développeurs débutants",
  "animateur" : {
    "prenom" : "Morgan",
    "nom" : "Friche",
    "mel" : "mfriche@mail.com",
    "telephone" : "06 39 98 65 14"
  },
  "nbPlaces" : 40,
  "materiels" : [
    {
      "libelle" : "ordinateur portable ",
      "quantite" : 1
    },
    {
      "libelle" : "IDE PHPStorm en mode évaluation gratuit",
      "quantite" : 1
    }
  ]
},
{
  "libelle" : "Introduction à la cybersécurité",
  "dateHeure" : "19/06/2021 16:00",
  "duree" : "2h",
  "salle" : "Alan Turing",
  "typePublic" : "étudiants ou jeunes développeurs débutants",
  "animateur" : {
    "prenom" : "Morgan",
    "nom" : "Friche",
    "mel" : "mfriche@mail.com",
    "telephone" : "06 39 98 65 14"
  },
  "nbPlaces" : 40,
  "materiels" : [
    {
      "libelle" : "bloc-notes papier",
      "quantite" : 1
    }
  ]
}
]
```

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 10 sur 18

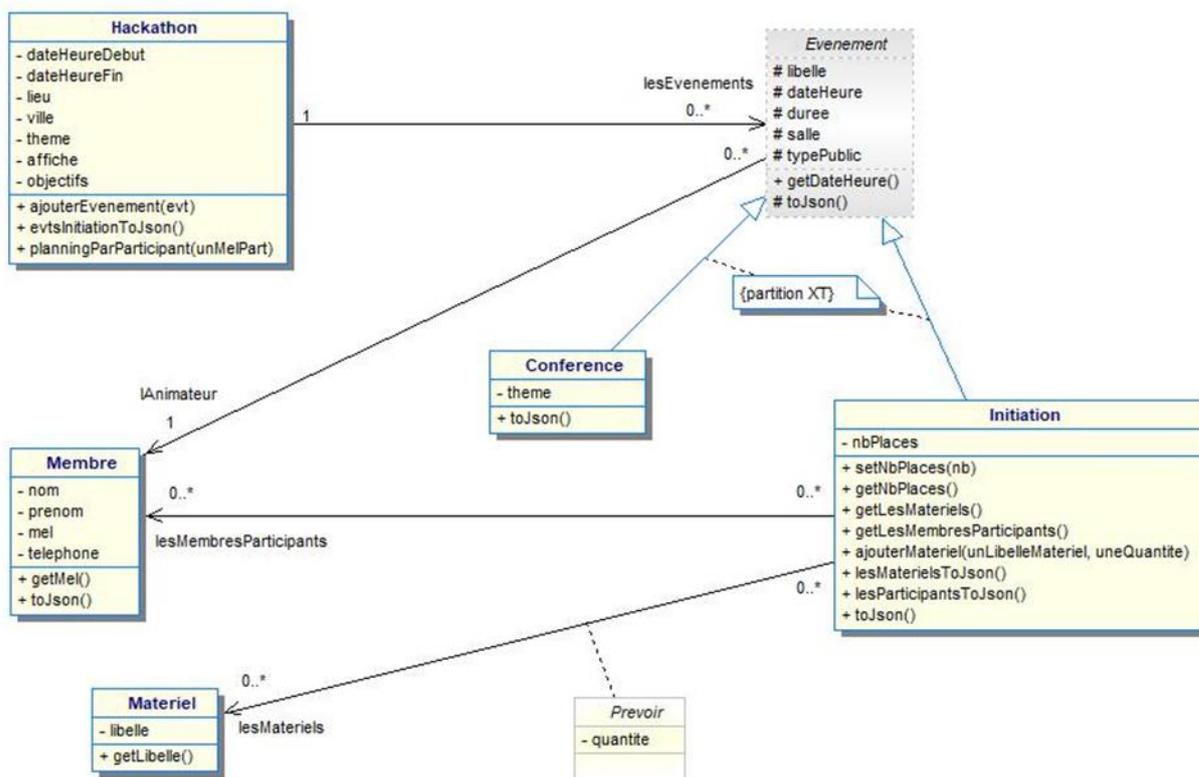
Exemple d'un événement de type conférence :

```

{ "libelle" : "Les méthodes agiles ?",
  "dateHeure" : "19/06/2021 09:00",
  "duree" : "2h",
  "salle" : "Hedy Lamarr",
  "typePublic" : "chefs de projets novices",
  "animateur" : {
    "prenom" : "Louison",
    "nom" : "Gelin",
    "mel" : "lgelin@mail.com",
    "telephone" : "06 39 98 23 01"
  },
  "theme" : "gestion de projet"
}

```

DOCUMENT 4 : diagramme de classes du module *Hackat'Event*



La classe Événement est une classe abstraite donc elle ne pourra pas être instanciée. Seules ses deux classes filles, Conférence et Initiation pourront être instanciées.

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 11 sur 18

DOCUMENT 5 : code *PHP* partiel des classes du module *Hackat'Event*

```
class Materiel{
    private $libelle;

    public function __construct($libM){
        $this->libelle = $libM;
    }
    public function getLibelle() {return $this-> libelle ;}
}
```

```
class Hackathon {
    private $dateHeureDebut;
    private $dateHeureFin;
    private $lieu;
    private $ville;
    private $theme;
    private $affiche;
    private $objectifs;
    private $lesEvenements; // type : collection d'objets Evenement

    public function __construct($dhd, $dhf, $lieu, $ville, $theme, $aff, $obj){
        $this->dateHeureDebut = $dhd;
        $this->dateHeureFin = $dhf;
        $this->lieu = $lieu;
        $this->ville = $ville;
        $this->theme = $theme;
        $this->affiche = $aff;
        $this->objectifs = $obj;
        $this->lesEvenements = array();
    }
    public function ajouterEvenement($evt){
        array_push($this->lesEvenements, $evt);
    }
    public function evtsInitiationToJson() { // retourne une chaine de caractères
        $chaineJson = "EvenementsInitiations" : [;
        foreach ($this->lesEvenements as $evt){
            if ($evt instanceof Initiation){
                $chaineJson = $chaineJson . $evt->toJson(); // concaténation
            }
        }
        return $chaineJson . " ]\n" ;
    }
    public function planningParParticipant($unMelPart) {
        $dicoEvtPart = array();
        foreach ($this->lesEvenements as $evt){
            if ($evt instanceof Initiation){
                foreach ($evt->getLesMembresParticipants() as $unMembre){
                    if ($unMembre->getMel() == $unMelPart){
                        $dicoEvtPart[$evt->getdateHeure()] = $evt;
                    }
                }
            }
        }
        return $dicoEvtPart;
    }
}
```

```

class Membre{
    private $nom;
    private $prenom;
    private $mel;
    private $telephone;

    public function __construct($nom, $prenom, $mel, $tel){...}
    public function getMel() { return $this->mel ; }
    public function toJson() {
        // méthode retournant tous les attributs de la classe au format JSON
        return json_encode(["nom"=> $this->nom,"prenom"=> $this->prenom,
            "mel"=>$this->mel,"telephone"=>$this->telephone]);
    }
}

```

```

abstract class Evenement{
    protected $libelle;
    protected $dateHeure;
    protected $duree;
    protected $salle;
    protected $IAanimateur; // type: Membre
    protected $typePublic;

    public function __construct($libelle, $dateHeure, $duree, $salle, $IAanimateur, $leTypePublic) {
        $this->libelle = $libelle;
        $this->dateHeure = $dateHeure;
        $this->duree = $duree;
        $this->sale = $salle;
        $this->IAanimateur = $IAanimateur;
        $this->typePublic = $leTypePublic;
    }
    public function getDateHeure(){ return $this->dateHeure;}
    protected function toJson(){
        // méthode retournant tous les attributs de la classe au format JSON
        return " { \"libelle\" : \""
            .$this->libelle."\", \n \"dateHeure\" : \""
            .$this->dateHeure."\", \n \"duree\" : \""
            .$this->duree."\", \n \"salle\" : \""
            .$this->salle."\", \n \"typePublic\" : \""
            .$this->typePublic."\", \n \"animateur\" : "
            .$this->IAanimateur->toJson();
    }
}

```

```

class Conference extends Evenement {
    private $theme;

    public function __construct($libelle, $dateHeure, $duree, $salle, $IAanimateur, $leTypePublic,
    $theme) {
        parent::__construct($libelle, $dateHeure, $duree, $salle, $IAanimateur, $leTypePublic);
        $this->theme = $theme;
    }
    public function toJson(){
        // méthode retournant tous les attributs de la classe au format JSON
        return parent::toJson().",\n \"theme\" : \"".$this->theme."\"\\n }\\n";
    }
}

```

```

class Initiation extends Evenement {
    private $nbPlaces;
    private $lesMateriels; /* Ce dictionnaire de (string, int) implémente la classe-association Prévoir.
                           La clé correspond au libelle du matériel et la valeur à la quantité.*/
    private $lesMembresParticipants; // type : collection de Membre

    public function __construct($libelle,$dateHeure, $duree, $salle, $IAnimateur,
    $leTypePublic, $nb) {
        // Question B.1 b) : à compléter et à reporter sur votre copie
    }
    public function setNbPlaces($nb){
        if ($nb >= 0) {
            $this->nbPlaces = $nb;
        }
    }
    public function getNbPlaces(){return $this->nbPlaces;}
    public function getLesMateriels(){return $this->lesMateriels;}
    public function getLesMembresParticipants(){
        return $this->lesMembresParticipants;
    }
    public function ajouterMateriel($unLibelleMateriel, $uneQuantite) {
        // Question B.1 a) : à corriger et à reporter sur votre copie
        if ($unLibelleMateriel != null ) {
            if (array_key_exists($unLibelleMateriel, $this->lesMateriels) == false)
            { // on vérifie que la clé n'existe pas déjà dans le dictionnaire
                $this->lesMateriels[$unLibelleMateriel] = $uneQuantite;
                /* ajoute dans le dictionnaire la clé de type String correspondant au libelleMateriel et la
                valeur de type entier correspondant à la quantité demandée*/
            }
        }
    }
    private function lesMaterielsToJson(){
        // retourne une chaine de caractères
        $chaineJson = "\"materiels\" : [ \n";
        $debutChaine = true;
        foreach ($this->lesMateriels as $lib => $qte) {
            if ($debutChaine == false) {
                // on va ajouter un élément supplémentaire, on le sépare par une virgule
                $chaineJson = $chaineJson . ",\n"; //concaténation
            }
            else {
                $debutChaine = false ;
            }
            // on ajoute l'élément (concaténation)
            $chaineJson = $chaineJson . "{ \n \"libelle\" : '\".$lib . "\",\n \"quantite\" : '\" . $qte . "\" }";
        }
        $chaineJson = $chaineJson . "] \n"; //concaténation
        return $chaineJson;
    }
    private function lesParticipantsToJson() { // retourne une chaine de caractères
        // Question B.1 d) : à compléter et à reporter sur votre copie
    }
    public function toJson() { // retourne une chaine de caractères
        // Question B.1 e) : à compléter et à reporter sur votre copie
    }
}

```

DOCUMENT 6 : extrait de la classe de test TestInitiation

```
class TestInitiation{

private $evtInitPHP ;

public function testCreationInitiation() {
    $IAnim = new Membre("Friche", "Morgan", "mfriche@mail.com", "06 39 98 23 01");
    $this->evtInitPHP = new Initiation("Introduction au PHP", "19/06/2021 13:00", "2h",
        "Alan Turing", $IAnim, "étudiants et jeunes développeurs", 40);
}

public function testAjouterMaterielQuantiteZero(){
    $this->testCreationInitiation(); // appel de la méthode qui instancie l'évènement initiation
    $leMateriel = new Materiel("ordinateur portable");
    $this->evtInitPHP->ajouterMateriel($leMateriel->getLibelle(), 0);
    $this->assertEquals(0, count($this->evtInitPHP->getLesMateriels()),
        "Matériel ajouté alors que la quantité est égale à 0");
}

public function testAjouterParticipantLimiteNombrePlaces() {
    $this->testCreationInitiation();
    // on force le nombre de places de l'initiation à 2 places
    $this->evtInitPHP->setNbPlaces(2);
    $leParticipant1 = new Membre("Mallien", "Yannick", "myannick@mail.com", "06 39 98 15 12");
    $this->evtInitPHP->ajouterParticipant($leParticipant1) ;
    $leParticipant2 = new Membre("Dus", "Dominique", "ddus@mail.com", "06 39 98 00 56 ");
    $this->evtInitPHP->ajouterParticipant($leParticipant2) ;
    $leParticipant3 = new Membre("Smith", "Jean", "jsmith@mail.com", "06 39 98 85 17");
    $this->assertFalse($this->evtInitPHP->ajouterParticipant($leParticipant3),
        "mauvaise gestion des places disponibles");
    $this->assertSame(2, count($this->evtInitPHP->getLesMembresParticipants()),
        "erreur dans l'ajout du 3ème participant");
}
}
```

DOCUMENT 7 : extrait de documentation PHP

Exemple d'utilisation de la collection (tableau indicé en PHP)

```
private $lesParticipants; // déclaration d'une variable d'instance qui pourra être une collection
$this->lesParticipants = array(); // création d'un tableau indicé de 0 à la taille du tableau (collection)
$unPart = new Participant(); // création d'une variable instance de la classe Participant
$nb = count($this->lesParticipants); // la variable nb récupère la taille du tableau $lesParticipants
array_push($this->lesParticipants, $unPart); // ajoute un participant à la fin du tableau
// $lesParticipants
foreach ($this->lesParticipants as $unParticipant) { // parcours de la collection
    echo $unParticipant->getMel(); // affiche le mél du participant courant
}
```

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 15 sur 18

Exemple d'utilisation du dictionnaire (tableau associatif en PHP)

```
private $lesMateriels ; // déclaration d'une variable d'instance qui pourra être un dictionnaire
                        // (tableau associatif)
$this->lesMateriels = array() ; // création d'un tableau associatif

$this->lesMateriels[$unLibelleMateriel] = $uneQuantite ; /* ajoute dans le dictionnaire la clé de type
chaîne de caractères correspondant au libelleMatériel et la valeur de type entier correspondant à
la quantité demandée*/

if ( array_key_exists($uneCle, $this->lesMateriels) == true) {
    // on vérifie que la clé existe déjà dans le dictionnaire
}
foreach ($this->lesMateriels as $cle => $valeur) {
    echo $cle; // affiche ici le libelleMateriel
    echo $valeur; // affiche ici la quantité nécessaire pour ce libellé de matériel
}
}
```

parent::méthode() permet d'appeler la méthode présente dans la classe mère.

instanceof permet de connaître le type d'instance d'un objet.

```
$unPart = new Participant(...);
if ($unPart instanceof Participant) {
    // traitement effectué si unPart est une instance de la classe Participant
}
}
```

DOCUMENT 8 : schéma relationnel de la base de données Hackat'Event

HACKATHON (id, dateHeureDebut, dateHeureFin, lieu, ville, theme, affiche, objectifs)
clé primaire : id

MEMBRE (id, nom, prenom, mel, telephone)
clé primaire : id

EVENEMENT (id, idHackathon, idAnimateur, libelle, dateHeure, duree, salle, typePublic)
clé primaire : id
clé étrangère : idHackathon en référence à id de Hackathon
clé étrangère : idAnimateur en référence à id de Membre

CONFERENCE (idEvenementConf, theme)
clé primaire : idEvenementConf
clé étrangère : idEvenementConf en référence à id de Evenement

INITIATION (idEvenementInit, nbPlaces)
clé primaire : idEvenementInit
clé étrangère : idEvenementInit en référence à id de Evenement

INSCRIRE (idEvenementInit, idMembre)
clé primaire : idEvenementInit, idMembre
clé étrangère : idEvenementInit en référence à idEvenementInit de Initiation
clé étrangère : idMembre en référence à id de Membre

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 16 sur 18

DOCUMENT 9 : interface graphique affichant la liste des hackathons

Date	Ville
12/06/2021	Paris
13/07/2021	Lilles
13/07/2021	Nice
13/07/2021	Orléans
14/08/2021	Bordeaux
14/08/2021	Brest
14/08/2021	Strasbourg

DOCUMENT 10 : fichier *liste.vue*

```
1. <template>
2.   <div>
3.     <div class="row" id="filtre">
4.       <div class="col">
5.         <input type="text" name="critere" placeholder="Votre critère de recherche..."
6.           v-model="recherche">
7.       </div>
8.     <div class="col">
9.       <input type="button" v-on:click="rechercher()">
10.    </div>
11.  </div>
12.  <div class="row">
13.    <div class="col text-center bigTitle">
14.      <!-- LIGNE À EXPLIQUER -->
15.      {{{lesHackathons.length}}}
16.    </div>
17.  </div>
18.  <!-- On affiche le résultat si la taille du tableau est > 0 -->
19.  <div v-if="lesHackathons.length > 0">
20.    <!-- Résultat -->
21.    <table>
22.      <tr>
23.        <th>Date</th>
24.        <th>Ville</th>
25.      </tr>
26.      <tr v-for="hackathon in lesHackathons" :key="hackathon.id">
27.        <td>{{hackathon.dateDebut}}</td>
28.        <td>{{hackathon.ville}}</td>
29.      </tr>
30.    </table>
31.  </div>
32.  <div v-else>
33.    Aucun résultat.
34.  </div>
35. </template>
```

DOCUMENT 11 : interface de programmation (API) *api-rechercher.php*

```
1. <?php
2. include("util/bdd.php");
3. header("Content-Type: application/json");
4. $pdo = connexionBdd(); // Connexion à la base de données
5. // Récupération avec filtrage du critère de recherche passé par la méthode POST
6. $critere = filter_input(INPUT_POST, 'critere', FILTER_SANITIZE_STRING);
7. // Requête de recherche.
8. $sql = 'SELECT DATE_FORMAT(dateHeureDebut, "%d/%m/%Y") AS dateDebut, ville
9. FROM HACKATHON WHERE ville LIKE :critere ORDER BY dateDebut, ville';
10. $stmt = $pdo->prepare($sql);
11. $stmt->bindValue(':critere', "%$critere%", PDO::PARAM_STR);
12. $stmt->execute();
13. // Récupération de l'ensemble des résultats
14. $rows = $stmt->fetchAll(PDO::FETCH_ASSOC);
15. echo json_encode($rows); // Affichage de la liste des hackathons
16. ?>
```

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS	SESSION 2021
U5 – Production et fourniture de services informatiques	Durée : 4 heures
Code sujet : SI5SLAM	Page 18 sur 18