

L'événement, appelé Innovation Days, vise à stimuler l'innovation et la créativité parmi les étudiants, les jeunes entrepreneurs et les passionnés du digital. Les participants devront développer des prototypes en 72 heures pour répondre à des défis spécifiques dans le domaine des technologies de l'information et de la communication.

POINTS CLÉS DU DOCUMENT ET FONCTIONNALITÉS :

Objectif du Hackathon :

- ❖ Stimuler l'innovation et la créativité.
- ❖ Appliquer les connaissances théoriques à des problèmes pratiques.
- ❖ Développer des prototypes en 72 heures.

Participants éligibles :

Étudiants de l'INPTIC.

- ❖ Étudiants d'autres établissements d'enseignement supérieur.
- ❖ Jeunes startups.
- ❖ Passionnés du digital.

Composition des équipes :

Chaque équipe doit être composée de 4 personnes avec des profils complémentaires :

- ❖ 2 développeurs.
- ❖ 1 technicien réseau.
- ❖ 1 marketeur.

Critères d'évaluation :

- ❖ Innovation (30%).
- ❖ Utilité (25%).
- ❖ Présentation (20%).
- ❖ Collaboration (15%).
- ❖ Prototype fonctionnel (10%).

Récompenses :

- ❖ Accès à un programme d'incubation ou d'accélération.
- ❖ Formations gratuites.
- ❖ Accès à des conférences.
- ❖ Voyages d'études (Learning Expeditions).
- ❖ Abonnements à des médias spécialisés.
- ❖ Cadeaux technologiques (tablettes, ordinateurs, montres connectées, etc.).

Logistique :

- ❖ Lieu : INPTIC (Feu rouge Gros bouquet).
- ❖ Dates : Du 23 au 25 Mars.
- ❖ Horaires : 8h00 à 18h00.
- ❖ Restauration fournie.

Support et mentorat disponibles.

Processus de soumission :

- ❖ Les équipes doivent soumettre un prototype fonctionnel accompagné d'une présentation détaillée.
- ❖ Un dossier de projet peut être requis, incluant des spécifications techniques, des diagrammes et une documentation du code source.

Suivi post-Hackathon :

- ❖ Les équipes gagnantes seront suivies et accompagnées dans le développement ultérieur de leur prototype.
- ❖ Enquête de satisfaction auprès des participants.
- ❖ Publication d'un rapport d'impact.

PROPOSITION DE FONCTIONNALITÉS

1. Module d'Inscription et de Gestion des Participants

Inscription des Participants :

- ✓ **Formulaire d'inscription pour les participants individuels.**
- ✓ **Validation des informations (âge, rôle, etc.).**
- ✓ **Envoi d'un email de confirmation après inscription.**

Gestion des Profils :

Espace personnel pour chaque participant.

- ✓ **Mise à jour des informations personnelles (email, téléphone, etc.).**
- ✓ **Téléchargement de justificatifs (carte d'étudiant, documents de startup, etc.).**

2. Module de Gestion des Défis et des Soumissions

Publication des Défis :

- ✓ **Interface pour les organisateurs pour publier les défis.**
- ✓ **Affichage des défis aux participants avec description et critères d'évaluation.**

Soumission des Projets :

- ✓ **Formulaire de soumission pour les équipes.**
- ✓ **Téléchargement des prototypes, présentations, et documentation.**
- ✓ **Validation des soumissions (format, taille des fichiers, etc.).**

Gestion des Soumissions :

- ✓ **Visualisation des soumissions par les juges et organisateurs.**
- ✓ **Attribution des soumissions aux juges pour évaluation.**

3. Module d'Évaluation et de Notation

Évaluation des Projets :

- ✓ **Interface pour les juges pour noter les soumissions.**
- ✓ **Calcul automatique des scores totaux basés sur les critères d'évaluation (innovation, utilité, présentation, collaboration, prototype fonctionnel).**
- ✓ **Commentaires et feedback des juges.**

Délibération du Jury :

- ✓ Interface pour les juges pour discuter et délibérer sur les projets.
- ✓ Sélection des gagnants basée sur les scores et discussions.
- ✓ Notification des gagnants par email.

Support Technique :

- ✓ Interface pour les participants pour demander de l'aide technique.
- ✓ Gestion des demandes par l'équipe technique.

7. Module de Logistique et de Planification

Gestion des Lieux et Horaires :

- ✓ Affichage des informations sur le lieu et les horaires du Hackathon.
- ✓ Gestion des créneaux horaires pour les sessions de mentorat et les ateliers.

Restauration :

- ✓ Gestion des repas et collations pour les participants.
- ✓ Inscription des participants pour les repas.

8. Module de Suivi Post-Hackathon

- ✓ Enquête de Satisfaction :
- ✓ Formulaire d'enquête pour les participants.
- ✓ Analyse des résultats pour améliorer les éditions futures.

Rapport d'Impact :

- ✓ Génération et publication du rapport d'impact du Hackathon.
- ✓ Affichage des réalisations et des innovations développées.

9. Module Administratif

Gestion des Utilisateurs :

- ✓ Création, modification, et suppression des comptes utilisateurs.
- ✓ Gestion des rôles (participant, juge, mentor, organisateur, administrateur).

Gestion des Données :

- ✓ Export des données (participants, équipes, soumissions, évaluations, etc.).
- ✓ Sauvegarde et restauration de la base de données.

Sécurité et Réglementation :

- ✓ Gestion des mesures de sécurité (authentification, autorisation, etc.).
- ✓ Respect de la propriété intellectuelle (gestion des droits d'auteur, etc.).

10. Module de Reporting et d'Analytique**Tableaux de Bord :**

- ✓ Tableaux de bord pour les organisateurs avec des statistiques sur les participants, les équipes, les soumissions, etc.
- ✓ Visualisation des scores et des évaluations.

Rapports :

- ✓ Génération de rapports sur les performances des équipes, les récompenses attribuées, etc.
- ✓ Export des rapports en PDF ou Excel.

1. Modèle Conceptuel de Données (MCD)

Voici une proposition des entités principales et leurs relations :

- **Participants** : Représente les participants individuels.
- **Équipes** : Représente les équipes formées par les participants.
- **Défis** : Représente les défis proposés lors du Hackathon.
- **Soumissions** : Représente les soumissions des équipes pour un défi.
- **Juges** : Représente les juges qui évaluent les soumissions.
- **Évaluations** : Représente les évaluations des juges pour chaque soumission.
- **Récompenses** : Représente les récompenses offertes aux équipes gagnantes.
- **Gagnants** : Représente les équipes gagnantes et les récompenses qu'elles ont reçues.
- **Mentors** : Représente les mentors qui assistent les équipes.
- **Sessions_Mentorat** : Représente les sessions de mentorat entre les mentors et les équipes.
- **Ressources** : Représente les ressources mises à disposition des participants.
- **Retours_Expérience** : Représente les retours d'expérience des participants.
- **Rapport_Impact** : Représente les rapports d'impact du Hackathon.

2. Modèle Logique de Données (MLD)

Le MLD détaille les tables, les attributs et les relations entre les tables.

Tables et Attributs :

Participants :

- **id_participant (PK)**
-
- **prenom**
-
- **nom**
-
- **email**
-
- **telephone**
-
- **institution**
-
- **role (ENUM: 'developpeur', 'technicien_reseau', 'marketeur')**
-
- **age**
-
- **date_inscription**

-
- **Équipes :**
-
- **id_equipe (PK)**
-
- **nom_equipe**
-
- **date_creation**

Membres_Équipe :

- **id_equipe (FK vers Équipes)**
-
- **id_participant (FK vers Participants)**
-
- **Défis :**
-
- **id_defi (PK)**
-
- **titre**
-
- **description**
-
- **date_annonce**
-
- **Soumissions :**
-
- **id_soumission (PK)**
-
- **id_equipe (FK vers Équipes)**
-
- **id_defi (FK vers Défis)**
-
- **description_prototype**
-
- **lien_presentation**
-
- **date_soumission**
-
- **Juges :**
-
- **id_juge (PK)**
-
- **prenom**
-
- **nom**

-
- **email**
-
- **expertise**
-

Évaluations :

- **id_evaluation (PK)**
-
- **id_soumission (FK vers Soumissions)**
-
- **id_juge (FK vers Juges)**
-
- **score_innovation**
-
- **score_utilite**
-
- **score_presentation**
-
- **score_collaboration**
-
- **score_prototype**
-
- **score_total**
-
- **commentaires**
-

Récompenses :

- **id_recompense (PK)**
-
- **nom_recompense**
-
- **description**
-
- **Gagnants :**
-
- **id_gagnant (PK)**
-
- **id_equipe (FK vers Équipes)**
-
- **id_recompense (FK vers Récompenses)**
-
- **date_attribution**

Mentors :

- **id_mentor (PK)**
-
- **prenom**
-
- **nom**
-
- **email**
-
- **expertise**

Sessions_Mentorat :

- **id_session (PK)**
-
- **id_equipe (FK vers Équipes)**
-
- **id_mentor (FK vers Mentors)**
-
- **date_session**
-
- **notes**

Ressources :

- **id_ressource (PK)**
-
- **nom_ressource**
-
- **lien_ressource**
-
- **description**

Retours_Expérience :

- **id_retour (PK)**
-
- **id_participant (FK vers Participants)**

-
- **commentaires**
-
- **note**
-
- **date_soumission**
-

Rapport_Impact :

- **id_rapport (PK)**
-
- **titre_rapport**
-
- **contenu_rapport**
-
- **date_publication**

-- Création de la base de données

CREATE DATABASE IF NOT EXISTS PlateformeHackathon;

USE PlateformeHackathon;

-- Table Participants

CREATE TABLE Participants (

id_participant INT AUTO_INCREMENT PRIMARY KEY,

prenom VARCHAR(50) NOT NULL,

nom VARCHAR(50) NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

telephone VARCHAR(20),

institution VARCHAR(100),

role ENUM('developpeur', 'technicien_reseau', 'marketeur') NOT NULL,

age INT CHECK (age >= 18),

date_inscription TIMESTAMP DEFAULT CURRENT_TIMESTAMP

);

-- Table Équipes

```
CREATE TABLE Équipes (  
id_equipe INT AUTO_INCREMENT PRIMARY KEY,  
nom_equipe VARCHAR(100) NOT NULL,  
date_creation TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- Table Membres_Équipe

```
CREATE TABLE Membres_Équipe (  
id_equipe INT,  
id_participant INT,  
PRIMARY KEY (id_equipe, id_participant),  
FOREIGN KEY (id_equipe) REFERENCES Équipes(id_equipe),  
FOREIGN KEY (id_participant) REFERENCES Participants(id_participant)  
);
```

-- Table Défis

```
CREATE TABLE Défis (  
id_defi INT AUTO_INCREMENT PRIMARY KEY,  
titre VARCHAR(200) NOT NULL,  
description TEXT,  
date_annonce TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- Table Soumissions

```
CREATE TABLE Soumissions (  
id_soumission INT AUTO_INCREMENT PRIMARY KEY,  
id_equipe INT,  
id_defi INT,  
description_prototype TEXT,
```

```
lien_presentation VARCHAR(255),  
date_soumission TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
FOREIGN KEY (id_equipe) REFERENCES Équipes(id_equipe),  
FOREIGN KEY (id_defi) REFERENCES Défis(id_defi)  
);
```

-- Table Juges

```
CREATE TABLE Juges (  
id_juge INT AUTO_INCREMENT PRIMARY KEY,  
prenom VARCHAR(50) NOT NULL,  
nom VARCHAR(50) NOT NULL,  
email VARCHAR(100) UNIQUE NOT NULL,  
expertise VARCHAR(100)  
);
```

-- Table Évaluations

```
CREATE TABLE Évaluations (  
id_evaluation INT AUTO_INCREMENT PRIMARY KEY,  
id_soumission INT,  
id_juge INT,  
score_innovation INT CHECK (score_innovation BETWEEN 0 AND 30),  
score_utilite INT CHECK (score_utilite BETWEEN 0 AND 25),  
score_presentation INT CHECK (score_presentation BETWEEN 0 AND 20),  
score_collaboration INT CHECK (score_collaboration BETWEEN 0 AND 15),  
score_prototype INT CHECK (score_prototype BETWEEN 0 AND 10),  
score_total INT,  
commentaires TEXT,  
FOREIGN KEY (id_soumission) REFERENCES Soumissions(id_soumission),  
FOREIGN KEY (id_juge) REFERENCES Juges(id_juge)  
);
```

-- Table Récompenses

```
CREATE TABLE Récompenses (  
id_recompense INT AUTO_INCREMENT PRIMARY KEY,  
nom_recompense VARCHAR(100) NOT NULL,  
description TEXT  
);
```

-- Table Gagnants

```
CREATE TABLE Gagnants (  
id_gagnant INT AUTO_INCREMENT PRIMARY KEY,  
id_equipe INT,  
id_recompense INT,  
date_attribution TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
FOREIGN KEY (id_equipe) REFERENCES Équipes(id_equipe),  
FOREIGN KEY (id_recompense) REFERENCES Récompenses(id_recompense)  
);
```

-- Table Mentors

```
CREATE TABLE Mentors (  
id_mentor INT AUTO_INCREMENT PRIMARY KEY,  
prenom VARCHAR(50) NOT NULL,  
nom VARCHAR(50) NOT NULL,  
email VARCHAR(100) UNIQUE NOT NULL,  
expertise VARCHAR(100)  
);
```

-- Table Sessions_Mentorat

```
CREATE TABLE Sessions_Mentorat (  
id_session INT AUTO_INCREMENT PRIMARY KEY,  
id_equipe INT,  
id_mentor INT,
```

```
date_session TIMESTAMP,  
notes TEXT,  
FOREIGN KEY (id_equipe) REFERENCES Équipes(id_equipe),  
FOREIGN KEY (id_mentor) REFERENCES Mentors(id_mentor)  
);
```

-- Table Ressources

```
CREATE TABLE Ressources (  
id_ressource INT AUTO_INCREMENT PRIMARY KEY,  
nom_ressource VARCHAR(100) NOT NULL,  
lien_ressource VARCHAR(255),  
description TEXT  
);
```

-- Table Retours_Expérience

```
CREATE TABLE Retours_Expérience (  
id_retour INT AUTO_INCREMENT PRIMARY KEY,  
id_participant INT,  
commentaires TEXT,  
note INT CHECK (note BETWEEN 1 AND 5),  
date_soumission TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
FOREIGN KEY (id_participant) REFERENCES Participants(id_participant)  
);
```

-- Table Rapport_Impact

```
CREATE TABLE Rapport_Impact (  
id_rapport INT AUTO_INCREMENT PRIMARY KEY,  
titre_rapport VARCHAR(200) NOT NULL,  
contenu_rapport TEXT,  
date_publication TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

Explications

- ❖ **Participants peuvent appartenir à une Équipe via la table Membres_Équipe.**
- ❖ **Équipes peuvent soumettre des Soumissions pour des Défis.**
- ❖ **Juges évaluent les Soumissions via la table Évaluations.**
- ❖ **Gagnants sont des Équipes qui ont gagné des Récompenses.**
- ❖ **Mentors peuvent avoir des sessions de mentorat avec des Équipes via Sessions_Mentorat.**
- ❖ **Ressources sont disponibles pour tous les participants.**
- ❖ **Retours_Expérience est collecté auprès des Participants après l'événement.**
- ❖ **Rapport_Impact est généré pour documenter les résultats et l'impact du Hackathon.**