

# PEC S2 - Réseau social lucratif

Matières	Intervenants
Suivi du projet	Karl Marques Bernardo
GoLang	Thomas Guillier
Flutter	Thomas Coichot

Dans un monde où les créateurs de contenu cherchent à monétiser leur communauté et leur savoir-faire, **OnlyFlick** se positionne comme une solution innovante qui permet aux utilisateurs de publier du contenu accessible via un abonnement premium. Cette plateforme offre aux créateurs la possibilité d'interagir avec leurs abonnés, de proposer des contenus exclusifs et de gérer des paiements sécurisés. Le projet vise à simuler le cycle complet du développement logiciel en environnement professionnel, tout en mettant l'accent sur la gestion de projet, l'architecture technique et le développement en équipe.

## Fonctionnalités principales /15

- **Inscription et Gestion des Profils**
  - Création de compte pour deux types d'utilisateurs : **Créateurs de contenu, Abonnés et Administrateur**
  - Gestion des informations personnelles, des préférences
- **Publication et Gestion de Contenu**
  - CRUD du contenu avec plusieurs visibilité (publique ou pour abonné)
  - Modération par l'administrateur
  - Possibilité de signaler du contenu
- **Système de Souscription et Paiement**
  - Mise en place d'abonnement premium et/ou de paiement unitaire
- **Interactions et Engagement**
  - Système de commentaires, de likes et de messages privés
- **Tableaux de Bord et Statistiques**
  - Tableau de bord créateur et Administrateur
    - Dashboard filtrable
      - Range de date
      - Dimensions
    - Kpis, Graphes, Top/Flop (pas que des count ou sum)

## Fonctionnalités supplémentaires /5

- Temps réel
- Notifications push
- Support multi-langues
- DeepLink / AppLink
- Etc, ... (sous validation)

## Contraintes par matière

- Golang
    - Tests unitaires
      - calcul prix panier minimum
    - Logs formatés
    - Pouvoir activer/désactiver une fonctionnalité
      - exemple type de produits
  - Flutter
    - Mobile (Android) et Web
    - Utilisation du système de thème
    - Responsive (Mobile et Web)
    - Tests intégrations
-

# Attendus RNCP

## Etude de faisabilité

- Etude de marché et analyse
- Identification des risques et contraintes
- Veille technologique : Comparaison des services Tiers (paiement, authentification, ...) et justification du choix

## Spécifications techniques et fonctionnelles

- Définition des acteurs (roles)
- Description des use cases (qui fait quoi et qui a le droit de)
  - workflow
  - matrice de droits

## Planification du projet

- Gestion d'équipe
  - Identification des profils nécessaires
  - Planning des réunions (avec preuve)
- Identification des coûts
- Système de gestion de projet : Planifié et réellement mise en place
- Répartition des tâches à chacun

## Conception technique du projet

- Schématisation BDD
- Mockup des interfaces

## Développement de la solution

- Application complète et fonctionnelle

## Tests et Mise en production

- Mise en place de tests (unitaires, minimum)
- Mise en place d'un CI/CD
- Déploiement en environnement de production
  - Full pour Golang
  - Génération d'APK et interface Web pour flutter

## Surveillance

- Formalisation des logs de sorties
  - Monitoring et alertes
    - Ex: Sentry, Crashlytics, Grafana
-

# Rendus attendus

- **Repository (github)**

**Vous devez TOUS commit sur votre repository** (*commits obligatoirement auto-signés*)

Un README.md

- noms de chacun et vos comptes github associés (pseudo)
- qui a fait quoi (listing des fonctionnalités du projet)
- etc...
- Procédure de lancement du projet en local

- **Mise en production**

Votre projet doit être obligatoirement mis en production pour le valider, le jour de la soutenance et ce durant 2 semaines au moins. Cette mise en production peut être sur n'importe quel type de serveur (VPS / dédié / heroku / raspberry / etc).

- **Cahier des charges**

- PDF dans le repository
- Golang
  - Repository Git
- Flutter
  - Repository Git + APK
- Sécurité
  - Aucune faille de sécurité autorisée (malus)

---

## Notation Matière vs PEC

### Matières

- **Contrôle.s continu.s** : Réalisé.s pendant le cours
- **Partiel** : Analyse technique sur le rendu

### PEC (Projet d'Étude de cas)

- **Contrôle continue** : Suivi de projet sur les différentes dates de suivi (avec votre responsable pédagogique)
- **Partiel** : Soutenance du PEC (rendu global)

## Barème sur la note de suivi du PEC (Projet d'Étude de cas)

Pour chaque session de suivi, le barème ci-dessous est à remplir pour chaque équipe par le responsable pédagogique. La note de suivi est une note de groupe, mais peut-être individualisée.

- Barème des suivis par groupe /20

Compétence à évaluer	Echec (0 pt)	Médiocre (1 pt)	Passable (2 pts)	Bien (3 pts)	Excellent (4 pt)
Présentation du projet (pitch rapide à l'oral du projet)					
Qualité du travail produit					
Respect du niveau de l'avancement du projet demandé					
Organisation du projet (attribution des tâches / etc)					
Cohésion du groupe					

---

## Organisation des soutenances

Pour soutenir votre projet, un ordre de passage sera réalisé, vous aurez 30 minutes par groupe découpées comme suit :

- 10 minutes de présentation de groupe
- 10 minutes d'entretien individuel

Template de présentation de votre projet (première partie) :

- Présentation rapide de votre concept (à l'oral, 1 minutes)
- Présentation des grandes fonctionnalités du projet (2 minutes max)
- Démonstration de votre projet à travers les différents parcours utilisateurs (7 minutes)

**Attention :** Aucune démonstration de code n'est demandée. Tenue professionnelle obligatoire. Projet lancé et prêt à être présenté (code source et maquette ouvert en fond pour les questions/réponses).

---