

UNIVERSITÉ NATIONALE DU VIETNAM À HANOÏ (UNVH)
INSTITUT FRANCOPHONE INTERNATIONAL (IFI)



Option : Systèmes Intelligents et Multimédia (SIM)

GESTION DE PROJET

**MISE EN PLACE D'UNE APPLICATION WEB ET
ANDROÏDE DE GESTION D'UNE FILE D'ACTUALITÉS**

Rapport rédigé par les étudiants :

MEDOU Daniel Magloire
KAFANDO Rodrique
SYLLA Aboubacar
BIAYA Emmanuel

Superviseur académique :

Dr HO Tuong Vinh (IFI)
Email : ho.tuong.vinh@ifl.edu.vn

PROMOTION XXI

Année académique : 2017-2018

Table des matières

1	Introduction	3
2	Présentation du projet	4
2.1	Objectifs du projet	4
2.2	Contexte	4
2.3	Enjeux	4
3	Analyse du projet	4
3.1	Exigences fonctionnelles	4
3.2	Le diagramme des cas d'utilisation	5
3.3	Exigences non fonctionnelles	5
4	Planning du projet	5
4.1	Découpage du projet en tâches	5
4.2	Estimations	7
4.3	Planification du projet dans le temps	8
5	Conception et implémentation de l'application	13
5.1	Architecture générale du système	13
5.2	Diagramme de classes	13
5.3	Diagramme de cas d'utilisation	13
6	Implémentation	13
6.1	Technologies et outils utilisés [1] [2]	13
6.2	Résultats obtenus et expérimentations	15
I	Conclusion Générale et Perspectives	28
1	Difficultés	28

Table des figures

1	Estimation de l'effort nécessaire pour le projet avec WBS	8
2	Les différents types d'acteurs de notre système	9
3	Calcul des cas d'utilisations ou exigences fonctionnelles	9
4	Calcul des cas d'utilisations ou exigences fonctionnelles	10
5	Calcul du facteur lié à l'environnement	11
6	Calcul final de l'effort en heures	11
7	Diagramme de GANTT	12
8	Diagramme WBS	12
9	Les couches de notre application	14
10	Le diagramme de classe	15
11	Le cas d'utilisation coté client	16
12	Le cas d'utilisation coté Administrateur	17
13	L'arborescence de notre application	18
14	Interface administrateur	19
15	Page d'accueil de la liste des actualités	19
16	Affichage du contenu d'une actualité	20
17	Interface d'accueil de Appyet	21
18	Première étape (définition du nom et du package)	21
19	Première étape (définition du nom et du package)	22
20	Liste des modules proposés par Appyet)	23
21	Interface pour la création du module)	24
22	Interface pour la création du module)	24
23	Interface pour la création du module)	25
24	Déroulement de l'application VNU)	26
25	Déroulement de l'application NEWS (site web de IFI))	26
26	Déroulement de IFI_ACTU (file d'actualité de IFI))	27

Liste des tableaux

1 Introduction

Vu la dimension internationale de l'institut Francophone International, il est nécessaire de nos jours qu'elle dispose un cadre d'échange d'informations qui pourrait en temps réel mettre à la disposition des internautes ou des utilisateurs les toutes nouvelles informations allant dans le sens de l'organisation administrative, culturelle, scientifique ou tout autres événements en son sein. C'est fort de ce constat et pour pallier à ce manque qu'il nous été confié la tâche de mise en place d'une application mobile intitulée **Gestion de File d'Actualité**. Ce projet relève du module Gestion de Projet qui est l'un des modules dispensé à l'IFI, le but est de renforcer les capacités des étudiants à pouvoir comprendre comment diriger des projets à partir de sa mise en place jusqu'à sa clôture. Afin de pouvoir mener à bien la mise en place du projet, nous organiserons le squelette de notre travail en quatre grands points. La première partie sera consacrée à une présentation de façon générale du projet, dans la deuxième partie nous effectuerons une analyse détaillée du sujet, dans la troisième partie nous parlerons de la conception et nous terminerons par la partie implémentation suivi d'une analyse des résultats obtenus.

2 Présentation du projet

Dans cette partie, nous présenterons tout d’abord les objectifs de notre application, ensuite nous situerons l’application dans son contexte et enfin nous parlerons des éventuels enjeux.

2.1 Objectifs du projet

L’essence de la conception de cette application a pour objectif principal d’offrir une vague d’informations d’actualités pour tout utilisateur à partir d’une application mobile. Notre application, une fois développée pourra être déployée et exécutée dans tout environnement ayant la capacité de démarrer un moteur de recherche. Tel est le cas des mobiles androïdes, pc, etc. Spécifiquement, cette dernière nous permettra d’afficher la liste d’actualités en utilisant le principe de scrolling, afficher le détail d’une actualité, gérer le mode déconnecté en utilisant le Local Storage.

2.2 Contexte

Aujourd’hui, il n’y a plus de doute possible sur l’importance d’avoir un site accessible par mobile qu’il soit responsive ou dédié. L’importance de cette technologie en primo consiste à augmenter la notoriété de la marque de l’entreprise, de rendre fidèle les utilisateurs, d’améliorer l’accessibilité, d’envoyer des notifications push, de créer des ponts entre le numérique et le physique, de pouvoir accéder aux informations même en étant déconnecté, de permettre une ouverture plus facile, et de détendre les utilisateurs. Au vu de tous ces avantages que présentent les applications mobiles, il est bien vrai que l’IFI dispose d’un site web à partir duquel il est possible d’accéder aux informations concernant son organisation et événements. Cependant, l’avènement des applications mobiles nous permet d’accéder à des services de tout genre tout en facilitant leur accès à partir des téléphones portables. Contrairement au site web où il faut passer par un navigateur et des recherches, l’application mobile nous permet d’y accéder par un simple clic sans compter des nombreux avantages précédemment cités.

2.3 Enjeux

Tout l’intérêt de notre application sera de fournir aux utilisateurs un moyen portatif qui leur permettra de recevoir et de consulter en temps réel les actualités de l’IFI. De plus, ce sera un outil efficace pour administration de l’IFI qui saura mettre en valeur son système de partage d’informations, même à l’endroit des utilisateurs les plus novices du domaine informatique. Un autre enjeu de notre système est qu’il peut être adapté, puis réutilisé par d’autres instituts pour la diffusion automatique de leurs informations.

3 Analyse du projet

3.1 Exigences fonctionnelles

Les fonctionnalités principales de notre application sont les suivantes :

1. Consulter l'ensemble des actualités en mode scrolling ;
2. Consulter le contenu d'une actualité à partir de son origine ;
3. Accéder en mode déconnecté les actualités via le principe du mode local storage (stockage locale) ;
4. Permettre à l'utilisateur d'ajouter ses liens favoris ;
5. Archiver les actualités déjà consultés ;
6. Supprimer une actualité

3.2 Le diagramme des cas d'utilisation

3.3 Exigences non fonctionnelles

Elles décrivent toutes les contraintes auxquelles doit répondre le système pour son bon fonctionnement et la fourniture d'un service de qualité. Dans notre cas, il s'agit de :

- **Ergonomie** : L'application doit offrir à l'utilisateur une interface conviviale et intuitive lui facilitant l'accès aux différentes fonctionnalités.
- **Rapidité** : L'application doit pouvoir répondre aux requêtes de l'utilisateur en un temps raisonnable.
- **Efficacité** : L'application doit fournir à l'utilisateur des informations fiables et vérifiables. Le système doit fournir des résultats utiles de façon aussi fine que possible.
- **Adaptabilité** : L'application doit être facilement adaptable pour d'autres institutions pour la gestion de leur file d'actualité.
- **Interactivité** : L'application doit interagir avec l'utilisateur pour faciliter la visualisation et l'exploitation des résultats produits.

4 Planning du projet

4.1 Découpage du projet en tâches

1. Conception
 - Interfaces utilisateur (UI)
 - Spécifications fonctionnelles (SF)
 - Production des diagrammes de cas d'utilisation (DCU)
 - Production des user story card pour les fonctionnalités prioritaires (USC)
 - Validation de la liste des fonctionnalités auprès des utilisateurs (VLF)
 - Spécifications techniques (ST)
 - Prise en main des technologies à utiliser (PTU)
 - Validation de la faisabilité du projet avec ces technologies (validation-PTU)
 - Conception de la base de données (CBD)
 - Réalisation du diagramme de classes (DC)
 - Réalisation du schéma relationnel (RSR).
 - Validation de la base de données (Validation-BD)

- Interfaces administrateur (AI)
 - Détermination des tâches nécessitant une administration (DTA)
 - Définition des groupes d'utilisateurs, des niveaux d'accès et des mécanismes de sécurité à mettre en place. (DGU)
 - Détermination des diagrammes à utiliser pour le reporting (DDR).
 - Conception des interfaces d'administration (CIA).
 - Rapports (RAPPORT)
 - Spécifications fonctionnelles (SF)
 - Définir les données nécessaires (DN)
 - Définir la fréquence de délivrance (FD)
 - Définir les utilisateurs susceptibles d'y accéder (DUS)
 - Délivrances de la liste complète des spécifications fonctionnelles et techniques du système (LCSF-T)
2. Développement (DEV)
- Interfaces Web (I-WEB)
 - Implémentation des différentes pages web (IMP-Pag -WEB)
 - Revue et validation de la dynamique fonctionnalités des interfaces (R&V-I)
 - Implémentation de la base de données (I-BD)
 - Implantation de la base de données dans un SGBD (I-SGBD)
 - Définition des contraintes sur les données (DC)
 - Mise en place de la couche d'accès aux données (MCAD)
 - Revue de la partie donnée via les interfaces (R-D-I)
 - Réalisation des outils de reporting (ROR)
 - Implémentation des diagrammes à l'aide d'outils statistiques (ID-OS)
 - Interconnexion des interfaces aux diagrammes (I-I-D)
 - Évaluation des résultats présentés sur les diagrammes en comparaison avec des résultats d'expérience manuelle (E-D)
 - Rapports (RAPPORT)
 - Implémentation des rapports intégrés au système (I-RIS)
 - Rédaction du rapport présentant les différentes phases du projet (R-RAPPORT)
 - Conduite des tests unitaires sur la base des user story card (T-Unit)
 - Revue et validation du rapport (R&VR)
3. Expérimentations (EXP)
- Interfaces web (I-WEB)
 - Vérification de la dynamique des interfaces (V-DI)
 - Tests d'intégration sur l'ensemble du système (TIS)
 - Expérimentations sur des données réelles (EDR)
 - Analyse des résultats d'expérimentation (ARE)
 - Interfaces d'administration (I-A)
 - Expérimentation de la sécurité d'accès (ESA)
 - Rapports (RAPPORT)

- Validation de la liste des fonctionnalités implémentées (V-FI)
- Rédaction du rapport final (RRF)
- 4. Training (TRNG)
 - Création d'un guide de déploiement(CGD)
 - Description de l'architecture du système(DAS)
 - Description de la procédure d'intégration des composants (DPIC)
 - Création d'un guide utilisateur (CGU)
 - Lister les fonctionnalités(LF)
 - Description de la manière d'y accéder (DMA)
 - Mise en œuvre d'un accès à l'aide au sein de l'application (HLP)
 - Revue et validation du guide utilisateur (RVGU).
- 5. Déploiement (DEP)
 - Matériels et logiciels (ML)
 - Mise en place des besoins matériels (BM)
 - Détermination des besoins logiciels (BL)
 - Déploiement de l'application (DPA)
 - Test et validation du déploiement
 - Création d'un guide de déploiement (GDEP)
 - Description de l'architecture du système (DAS)
 - Description de la procédure d'intégration des composants (DPIC)
 - Élaboration du plan de déploiement (EPD)
 - Rédaction du guide de déploiement (RGD)
 - Revue et validation du guide de déploiement (RVGD)

4.2 Estimations

Pour l'estimation de l'effort nécessaire de notre projet, nous avons utilisé deux méthodes que sont le WBS et la méthode des Use Cases Points.

1. La méthode WBS Nous présentons dans la figure 26 une estimation de l'effort à fournir pour compléter notre projet dans les délais impartis. Cette estimation a été faite à l'aide de la méthode WBS.

L'estimation de l'effort à partir de wbs se fait en définissant les valeurs des proportions de cas, c-à-d le pire des cas, le cas moyen, et le meilleur des cas. Les valeurs de chaque cas sont évalués en terme de jour, heure ou année pour calculer le SD value en tenant compte de la valeur de confiance préalablement définit, soit à 50%, 70%, 90%. Dans notre cas, nous avons calculé l'effort nécessaire pour la réalisation de notre projet en terme de jour et avec une valeur de confiance à 90%.

2. La méthode d'estimation par Use Cases Points Le méthode Use Cases Points permet de calculer le coût prévisionnel de l'effort de notre application selon les aspects suivants :
 - Le niveau d'implication des types d'acteurs intervenant dans la réalisation du projet.

B	C	D	E	F	G	H
Features / Functions	Best Case Estimate a	Most Likely Estimate m	Worst Case Estimate b	E Value	SD Value	Task Estimate 95% Confidence
Saisir une actualité	1	3	4	2,9	0,5	3,9
Ajouter une actualité	1	2	4	2,2	0,5	3,2
Rechercher une actualité	1	2	4	2,2	0,5	3,2
Afficher le contenu d'une actualité	1	3	6	3,2	0,9	5,0
Afficher la file d'actualité par page	1	2	5	2,4	0,7	3,8
Afficher la file d'actualité en utilisant le principe de scrolling	1	3	5	3,0	0,7	4,4
Archiver les actualités déjà consultés	1	2	4	2,2	0,5	3,2
Editer une actualité	1	2	4	2,2	0,5	3,2
Consulter le contenu d'une actualité	1	2	4	2,2	0,5	3,2
Consulter la liste des actualités présente dans la file	1	3	4	2,9	0,5	3,9
Supprimer une actualité	1	2	4	2,2	0,5	3,2
Totals	11	26	48	28	7,0	41
E (Project Work)	28,0					
SD (Project Work)	7,00					
Project Estimate > 95% Confidence	42					
NB: DANS LE CADRE DE NOTRE PROJET, IL SERA ESTIME EN TERME DE JOUR						

FIGURE 1 – Estimation de l'effort nécessaire pour le projet avec WBS

- La catégorisation du niveau de complexité des différentes exigences fonctionnelles pour chaque acteur donné.
- Le coût de l'effort intrinsèque est par la suite calculé en fonction des facteurs environnementaux, techniques et l'indice de multiplicité.

4.3 Planification du projet dans le temps

1. Le diagramme de GANTT Pour pouvoir exécuter notre projet de la meilleure façon possible, nous utilisons le diagramme de GANTT. Le diagramme de Gantt est un outil permettant de visualiser dans le temps les diverses tâches du projet. Il permet de représenter graphiquement l'avancement du projet. Dans notre diagramme de Gantt on représente :
 - En abscisse les unités de temps (exprimées en mois, en semaine ou en jours)
 - En ordonnée les différents postes de travail (ou les différentes tâches)
 La durée d'utilisation d'un poste de travail (ou la durée d'exécution d'une tâche) est matérialisée par une barre horizontale. Il est également fréquent de matérialiser par des flèches, les liens de dépendance entre les tâches (la flèche relie la tâche précédente à la tâche suivante). Donc pour visualiser ces tâches, nous utilisons la figure suivante que nous avons mise en place au démarrage de notre projet.
2. Le diagramme WBS La construction du diagramme WBS (figure 8) est une technique permettant de faire la planification des différentes tâches dans le temps et définir le coût des chacune de ces tâches de façon hiérarchique. De plus, il permet de définir la priorité de réalisation des tâches les unes après les autres.

Actor Summary		Multiplier	Number of Actors	Description
1	Simple	1	0	Simple actors are other systems that communicate with your software via a pre-defined API. An API could be exposed through a dll, or as a REST, SOAP, or any web-service API or remote procedure call (RPC). The key element is that you are exposing interaction with your software through a specific, well-defined mechanism.
2	Average	2	0	Average actors can either be human beings interacting in a well defined protocol, or they could be systems that interact through a more complex or flexible API.
3	Complex	3	2	The original definition of complex actors specifies that users who interact with the software through a graphical user interface are complex actors. While that is true, the same classification should apply to users who interact with the system in unpredictable ways. An AJAX interface that exposes more of the underlying application (and data stores) than would be available through a rigid protocol might introduce similar complexity.
			6	
Individual Actors		Multiplier	Actor Name	
1	Complex	3	Administrateur	
2	Simple	1		
3	Complex	3	Utilisateur	
4	Simple	1		
5	Simple	1		
6	Simple	1		
7	Simple	1		
8	Simple	1		
Insert additional rows above this row and copy the cell values to automatically update the counts of actors by type				

FIGURE 2 – Les différents types d'acteurs de notre système

Unadjusted Use Case Points		Multiplier	Number of Use Cases	Description
1	Simple	5	6	Simple Use Case - <u>up to</u> 3 transactions.
2	Average	10	5	<u>Average</u> Use Case - 4 <u>to</u> 7 transactions.
3	Complex	15	0	<u>Complex</u> Use Case - more <u>than</u> 7 transactions.
Calculated UUCP			80	
Individual Use Cases		Multiplier	Use Case Name	
1	Simple	5	Saisir une actualité	
1	Simple	5	Ajouter une actualité	
1	Simple	5	Rechercher une actualité	
2	Average	10	Afficher le contenu d'une actualité	
1	Simple	5	Afficher la file d'actualité par page	
1	Simple	5	Afficher la file d'actualité en utilisant le principe de scrolling	
1	Simple	5	Archiver les actualités déjà consultés	
2	Average	10	Supprimer une <u>actualiter</u>	
2	Average	10	Editer une actualité	
3	Average	10	Consulter le contenu d'une actualité	
3	Average	10	Consulter la liste des actualités présente dans la file	
Insert additional rows above this row and copy the cell values to automatically update the counts of actors by type				

FIGURE 3 – Calcul des cas d'utilisations ou exigences fonctionnelles

Technical Factor		Multiplier	Relative Magnitude (Enter 0-5)	Description
1	<u>Distributed System Required</u>	2	0	The architecture of the solution may be centralized or single-tenant, or it may be distributed (like an n-tier solution) or multi-tenant. Higher numbers represent a more complex architecture.
2	<u>Response Time Is Important</u>	1	0	The quickness of response for users is an important (and non-trivial) factor. For example, if the server load is expected to be very low, this may be a trivial factor. Higher numbers represent increasing importance of response time (a search engine would have a high number, a daily news aggregator would have a low number).
3	<u>End User Efficiency</u>	1	1	Is the application being developed to optimize on user efficiency, or just capability? Higher numbers represent projects that rely more heavily on the application to improve user efficiency.
4	<u>Complex Internal Processing Required</u>	1	1	Is there a lot of difficult algorithmic work to do and test? Complex algorithms (resource leveling, time-domain systems analysis, OLAP cubes) have higher numbers. Simple database queries would have low numbers.
5	<u>Reusable Code Must Be A Focus</u>	1	1	Is heavy code reuse an objective or goal? Code reuse reduces the amount of effort required to deploy a project. It also reduces the amount of time required to debug a project. A shared library function can be re-used multiple times, and fixing the code in one place can resolve multiple bugs. The higher the level of re-use, the lower the number.
6	<u>Installation Ease</u>	0,5	1	Is ease of installation for end users a key factor? The higher the level of competence of the users, the lower the number.
7	<u>Usability</u>	0,5	1	Is ease of use a primary criteria for acceptance? The greater the importance of usability, the higher the number.
8	<u>Cross-Platform Support</u>	2	1	Is multi-platform support required? The more platforms that have to be supported (this could be browser versions, mobile devices, etc. or Windows/OSX/Unix), the higher the value.
9	<u>Easy To Change</u>	1	1	Does the customer require the ability to change or customize the application in the future? The more change / customization that is required in the future, the higher the value.
10	<u>Highly Concurrent</u>	1	1	Will you have to address database locking and other concurrency issues? The more attention you have to spend to resolving conflicts in the data or application, the higher the value.
11	<u>Custom Security</u>	1	1	Can existing security solutions be leveraged, or must custom code be developed? The more custom security work you have to do (field level, page level, or role based security, for example), the higher the value.
12	<u>Dependence On Third-Party Code</u>	1	0	Will the application require the use of third party controls or libraries? Like re-usable code, third party code can reduce the effort required to deploy a solution. The more third party code (and the more reliable the third party code), the lower the number.
13	<u>User Training</u>	1	1	How much user training is required? Is the application complex, or supporting complex activities? The longer it takes users to cross the suck threshold (achieve a level of mastery of the product), the higher the value.
Calculated TCF			0,7	

FIGURE 4 – Calcul des cas d'utilisations ou exigences fonctionnelles

Environmental Factor		Multiplier	Relative Magnitude (Enter 0-5)	Description
1	Familiarity With The Project	1,5	3	How much experience does your team have working in this domain? The domain of the project will be a reflection of what the software is intended to accomplish, not the implementation language. In other words, for an insurance compensation system written in java, you care about the team's experience in the insurance compensation space - not how much java they've written. Higher levels of experience get a higher number.
2	Application Experience	0,5	3	How much experience does your team have with the application. This will only be relevant when making changes to an existing application. Higher numbers represent more experience. For a new application, everyone's experience will be 0.
3	OO Programming Experience	1	3	How much experience does your team have at OO? It can be easy to forget that many people have no object oriented programming experience if you are used to having it. A user-centric or use-case-driven project will have an inherently OO structure in the implementation. Higher numbers represent more OO experience.
4	Lead Analyst Capability	0,5	3	How knowledgeable and capable is the person responsible for the requirements? Bad requirements are the number one killer of projects - the Standish Group reports that 40% to 60% of defects come from bad requirements. Higher numbers represent increased skill and knowledge.
5	Motivation	1	3	How motivated is your team? Higher numbers represent more motivation.
6	Stable Requirements	2	3	Changes in requirements can cause increases in work. The way to avoid this is by planning for change and instituting a timing system for managing those changes. Most people don't do this, and some rework will be unavoidable. Higher numbers represent more change (or a less effective system for managing change).
7	Part Time Staff	-1	3	Note, the multiplier for this number is negative. Higher numbers reflect team members that are part time, outside consultants, and developers who are splitting their time across projects. Context switching and other intangible factors make these team members less efficient.
8	Difficult Programming Language	-1	3	This multiplier is also negative. Harder languages represent higher numbers. We believe that difficulty is in the eye of the be-coder (groan). Java might be difficult for a fortran programmer. Think of it in terms of difficulty for your team, not abstract difficulty.
Calculated EF			0,995	

FIGURE 5 – Calcul du facteur lié à l'environnement

Calculations From Other Tabs		
TCF	Technical Complexity Factor	0,7
EF	Environmental Factor	0,995
UUCP	Unadjusted Use Case Points	80
AW	Actor Weighting	6
Calculation of Use Case Points		
UCP	Use Case Points	59,9
Calculation of Estimated Effort		
Ratio	Hours of Effort per Use Case Point	20
Hours of Effort		1 198

FIGURE 6 – Calcul final de l'effort en heures

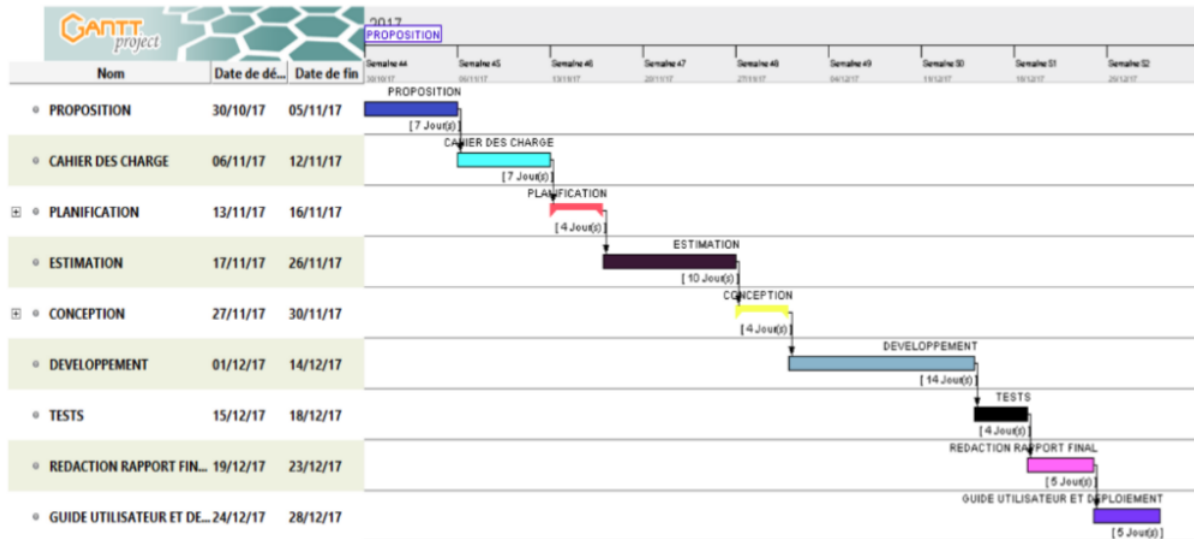


FIGURE 7 – Diagramme de GANTT

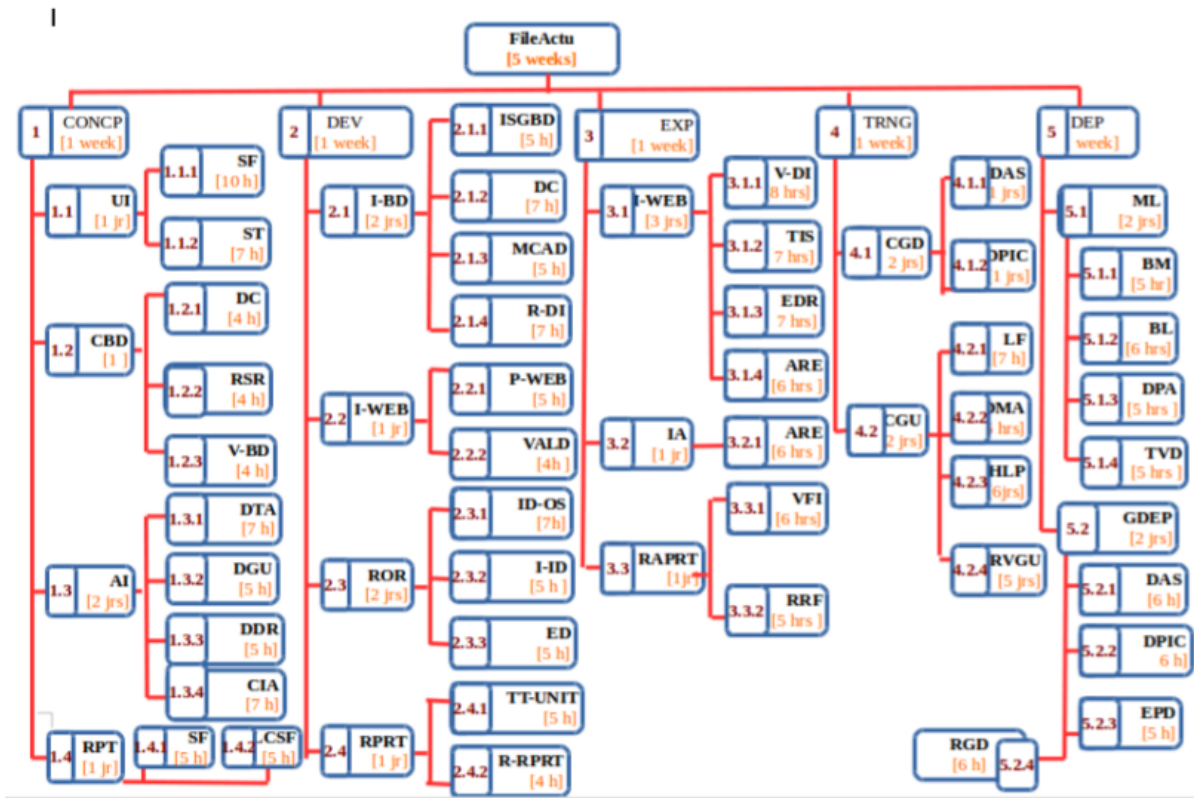


FIGURE 8 – Diagramme WBS

5 Conception et implémentation de l'application

5.1 Architecture générale du système

Afin d'offrir à notre projet une structure de qualité, nous avons opté pour l'architecture en 5 couches décrites comme suit :

1. **La couche Physique**

Cette couche correspond à la structure physique des données en d'autre terme la base de données de notre application.

2. **La couche Mapping**

Elle permet de représenter chaque table de la couche physique par une classe ainsi que de définir les fonctions comme les accesseurs et les mutateurs pour chaque classe. Elle correspond au package Model.Pojo dans notre application.

3. **La couche Entreprise**

Cette couche permet de définir pour chaque classe de la couche Mapping et les méthodes permettant de réaliser les fonctionnalités d'ajout, de modification, de suppression, de recherche et de sélection. Cette couche correspond au package Model.Dao dans notre application.

4. **La couche Application**

Elle joue le rôle de contrôleur et permet de vérifier si les informations entrées par l'utilisateur sont correctes. Si oui, elle fait appel à la méthode ou à la fonction correspondante à cette action au niveau de la couche Entreprise. Cette couche correspond au package Controller.Admin dans notre application.

5. **La couche Client**

Elle regroupe toutes les vues de notre application. Elle correspond au dossier Web Pages dans notre application.

5.2 Diagramme de classes

Notre application a été développé selon le diagramme de classe présenté dans la figure 10.

5.3 Diagramme de cas d'utilisation

Dans cette partie, nous illustrons principalement deux cas deux d'utilisations. Le cas d'utilisation coté client figure 11 et coté administrateur figure 12.

6 Implémentation

6.1 Technologies et outils utilisés [1] [2]

Pour la réalisation du projet « File-Actu », l'équipe de développement aura besoin des outils ci-après :

Description des couches	Exemple de File_Actu
La couche DAO , est en charge de la gestion des relations avec les sources de données, elle offre des services d'accès, de recherche, de création, de mise à jour, de suppression de données stockées dans une base de données. Cette couche s'appuie sur une interface standard qui est JPA (Java Persistence API) qui permet de masquer l'implémentation réelle du moteur de persistance qui est Hibernate.	utilisation d'une base de données relationnelle
La couche service , permet l'implémentation de l'ensemble de la logique métier de l'application, indifféremment des sources de données utilisées et de la présentation. Elle s'appuie sur les couches DAO et model pour effectuer des opérations CRUD (Create, Research, Update, Delete) sur des objets persisté et leur appliquer ensuite des traitements métier.	Objet : Actualité. Service : Ajouter les actualités
La couche Web , c'est la couche à laquelle a accès le client. C'est dans cette couche que sont créés les différents contrôleurs et les vues qui seront affichées à l'utilisateur dans un navigateur web par l'intermédiaire d'un serveur web comme TomCat.	Service en ligne de l'application : accès aux Actualités requête client : seront gérées par le protocole HTTP et l'affichage par HTML
La couche utilisateur , c'est la couche qui est chargée de gérer l'interface utilisateur. C'est à partir de cette interface que l'utilisateur peut effectuer ses opérations.	Consulter les actualités

FIGURE 9 – Les couches de notre application

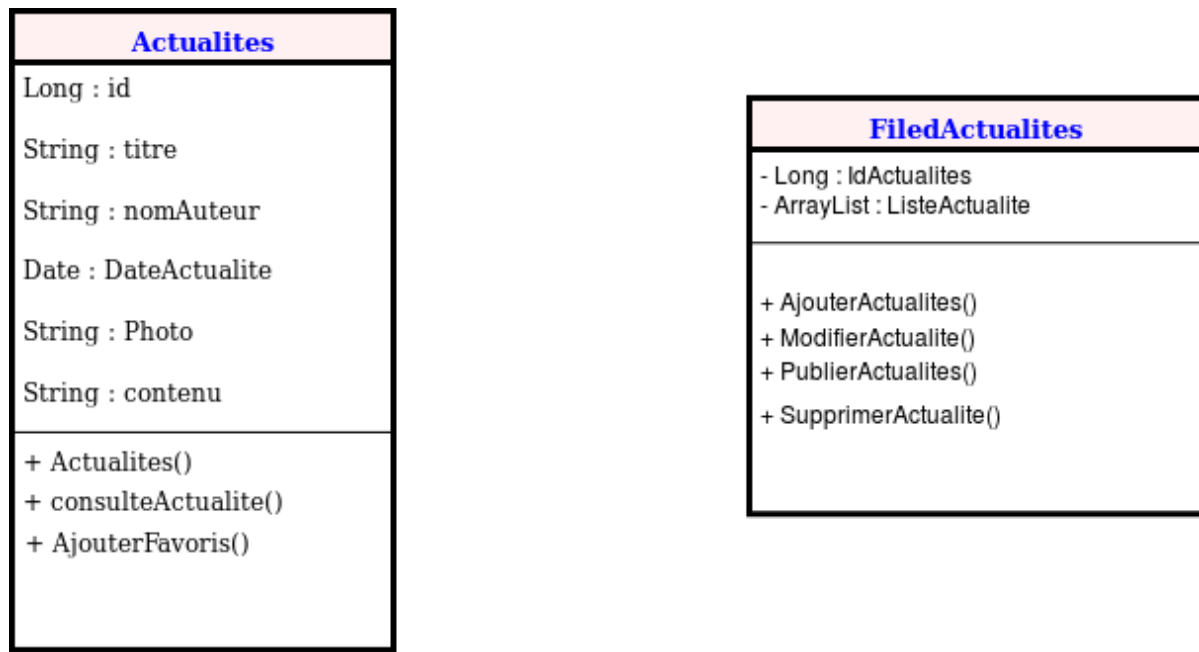


FIGURE 10 – Le diagramme de classe

- IDE Eclipse, environnement de développement partie back end
- IDE Web Storm, environnement de développement partie front end
- Linux 16.04, comme système d'exploitation
- NodeJS
- Framework Spring Boot
- Angular
- Ionic
- Cordova
- Androïde SDK Tools
- Gantt Project, logiciel de planification de projet
- Bootstrap, Framework css
- PhpMyAdmin, comme SGBD

6.2 Résultats obtenus et expérimentations

Pour la phase de réalisation, nous avons mise place notre application dans deux cas. Le premier cas à consisté à la mise en place à la file d'actualité avec les technologies précédemment cités tandis que le second cas a été mise en place à partir d'une plate forme prédéfinie du nom **Appyjet**.

1. Premier cas

Dans ce premier cas, notre application fonctionne à partir d'un ensemble d'informations mise à jour de façon régulière par un administrateur. L'utilisateur accède à la file d'actualités et peut consulter le contenu de chacune de ces actualités. Les captures suivantes

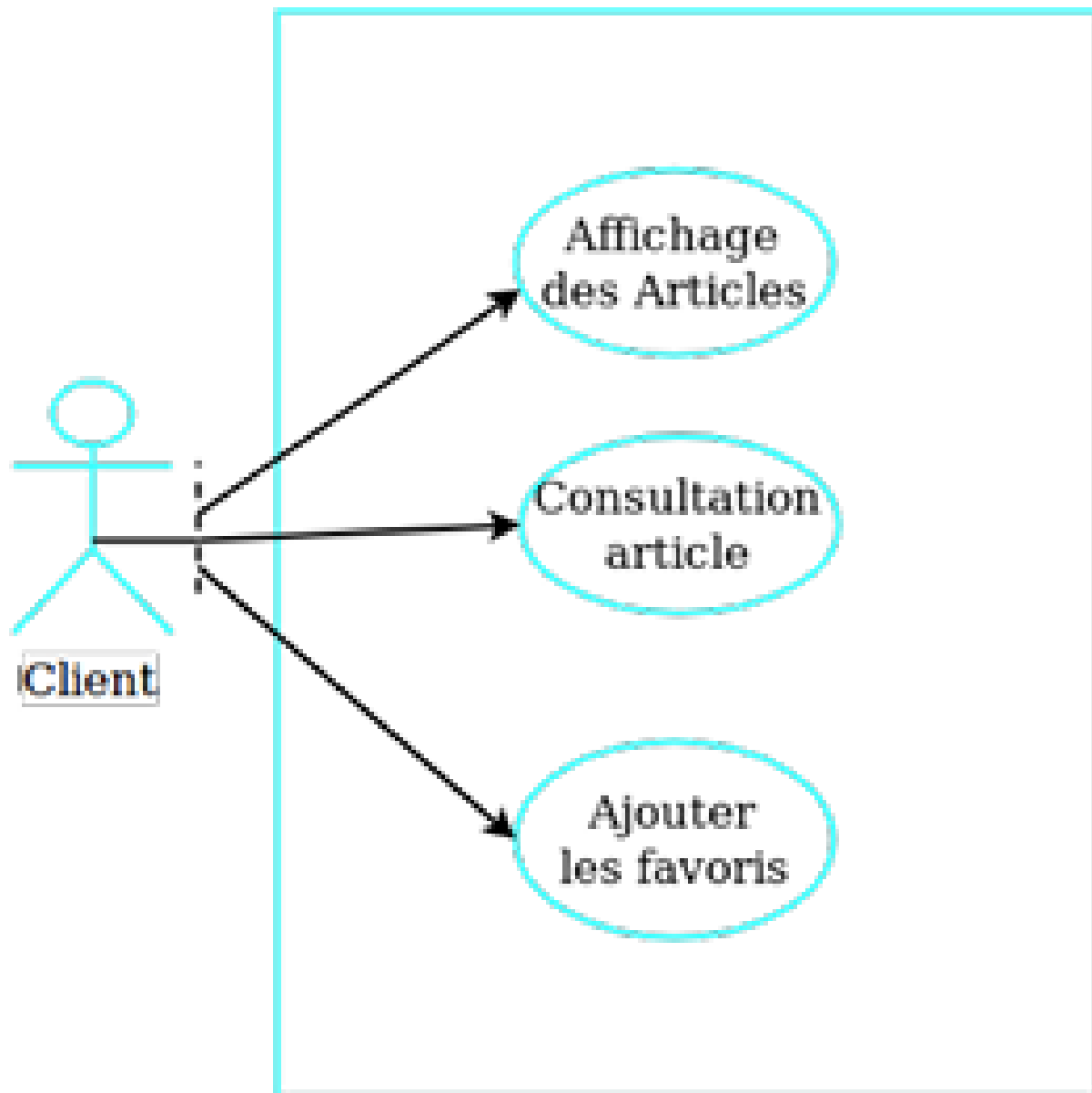


FIGURE 11 – *Le cas d'utilisation coté client*

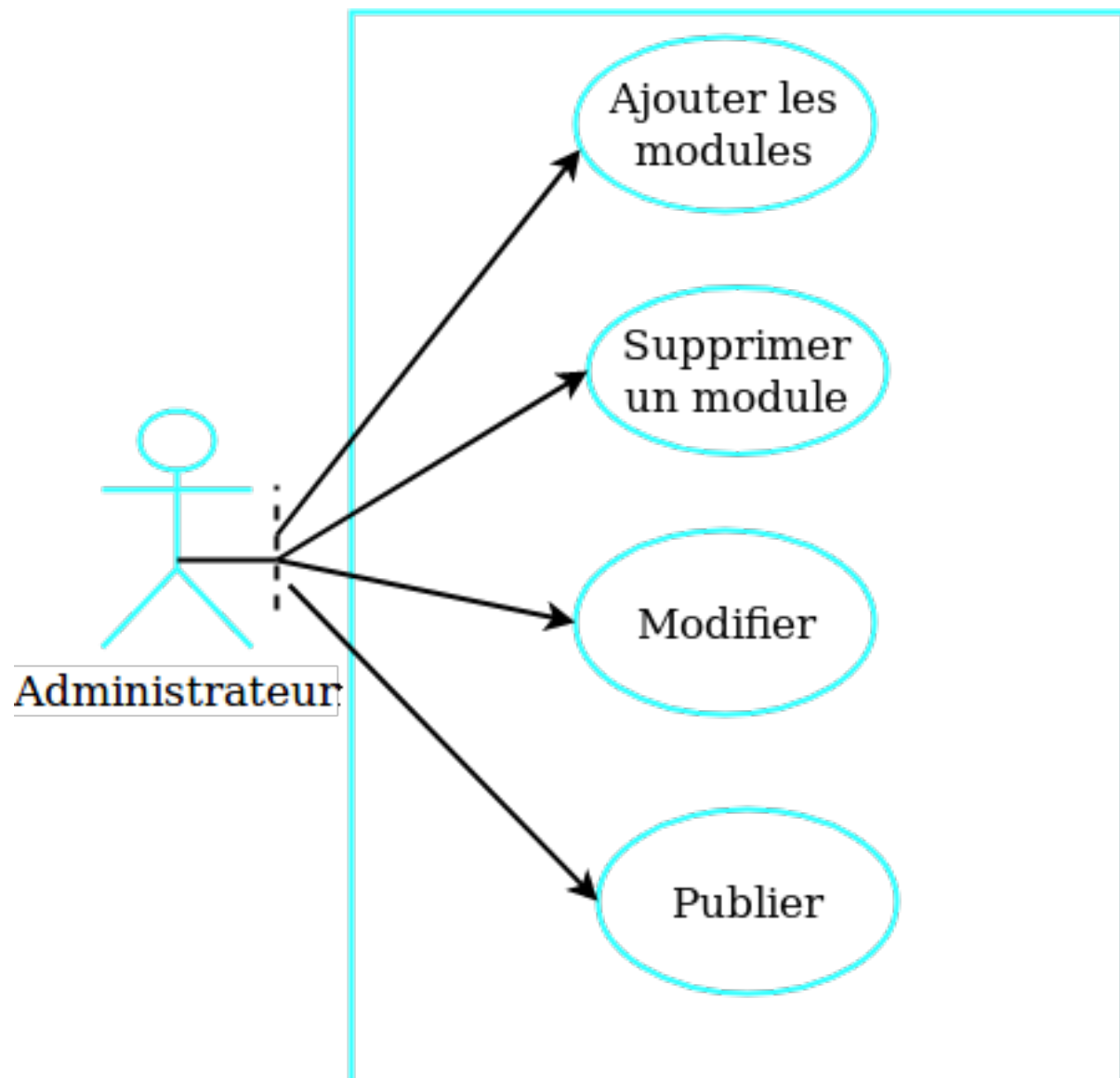


FIGURE 12 – Le cas d'utilisation coté Administrateur

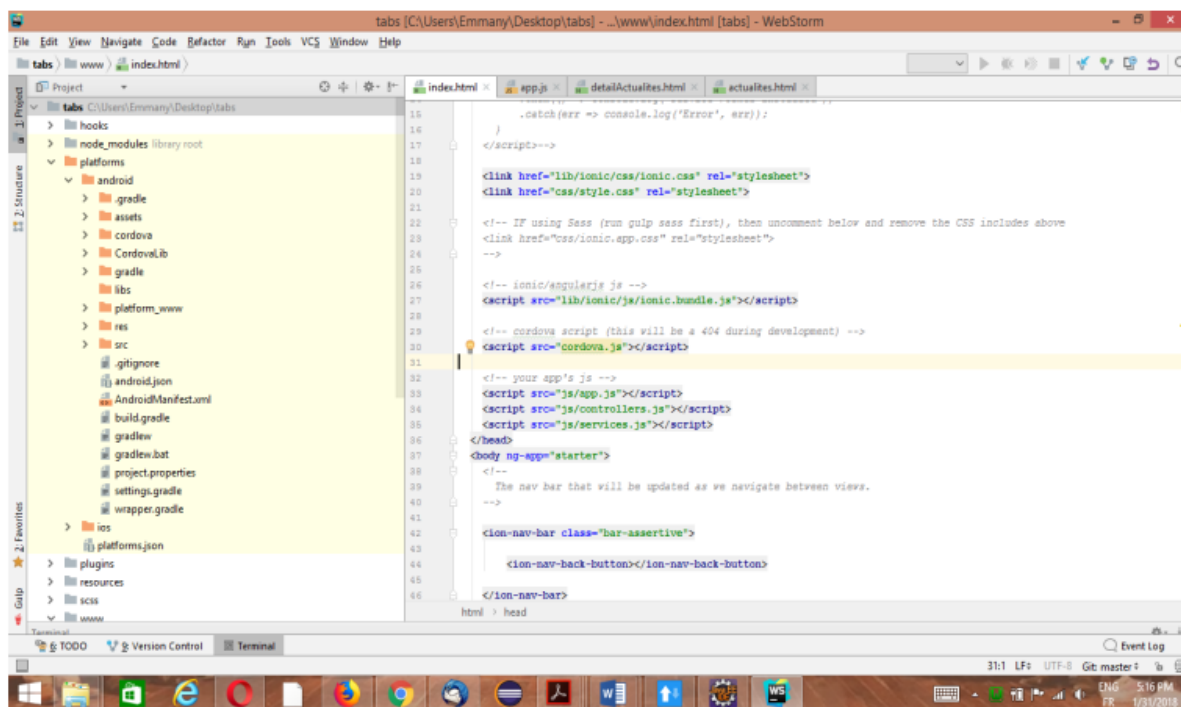


FIGURE 13 – L'arborescence de notre application

nous permettent de visionner les résultats obtenus. Nous pouvons illustrer :

- L'arborescence du contenu de notre application

Les différents fichiers que nous avons utilisés peuvent être représentés comme indiqué dans la figure 13.

- L'interface administrateur Il s'agit de l'interface à partir de laquelle l'administrateur met à jour la file d'actualité. Nous avons utilisé un outil ARC (Advanced REST Client) qui permet d'exécuter des requêtes POST. La figure 14 nous permet de voir l'interface et un exemple de commande d'insertion.

- Affichage de la liste des actualité

Une fois la requête POST exécutée, nous pouvons afficher la file d'actualité comme présenté à la figure 15.

- Accès au contenu d'une file d'actualité L'utilisateur peut accéder au contenu de chaque actualité en cliquant juste sur l'article. Illustration à la figure 16

2. Second cas

Dans cette partie, pour la mise en place de notre application, nous avons trouvé un outils du nom Appyet [3]. En effet, Appyet est un outil possédant une version gratuite et idéal permettant de créer une application mobile native pour Androïde dans un temps réduit. Facile d'utilisation, il ne nécessite pas de connaissance spécifique en programmation. Appyet dispose des modules prédéfinis permettant de créer une succession de pages.

Pour utiliser appyet, l'utilisateur doit créer un compte d'utilisateur à partir du quel il devrait s'identifier pour avoir accès aux différents services. La figure ci-dessous nous présente

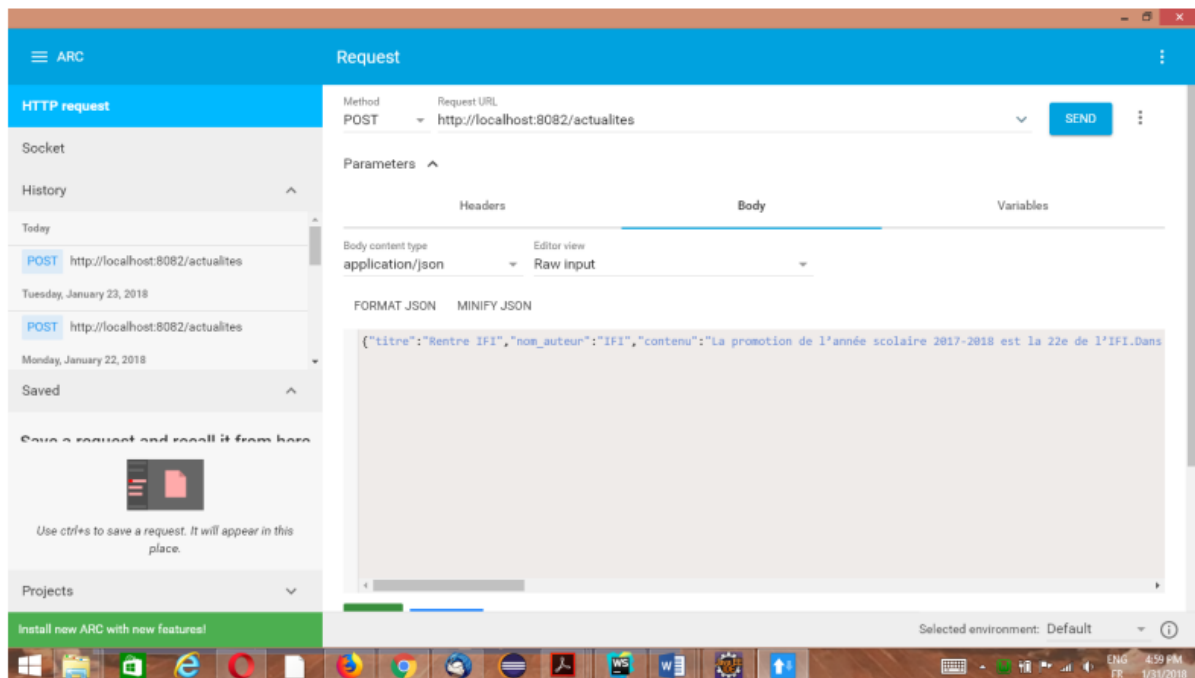


FIGURE 14 – Interface administrateur

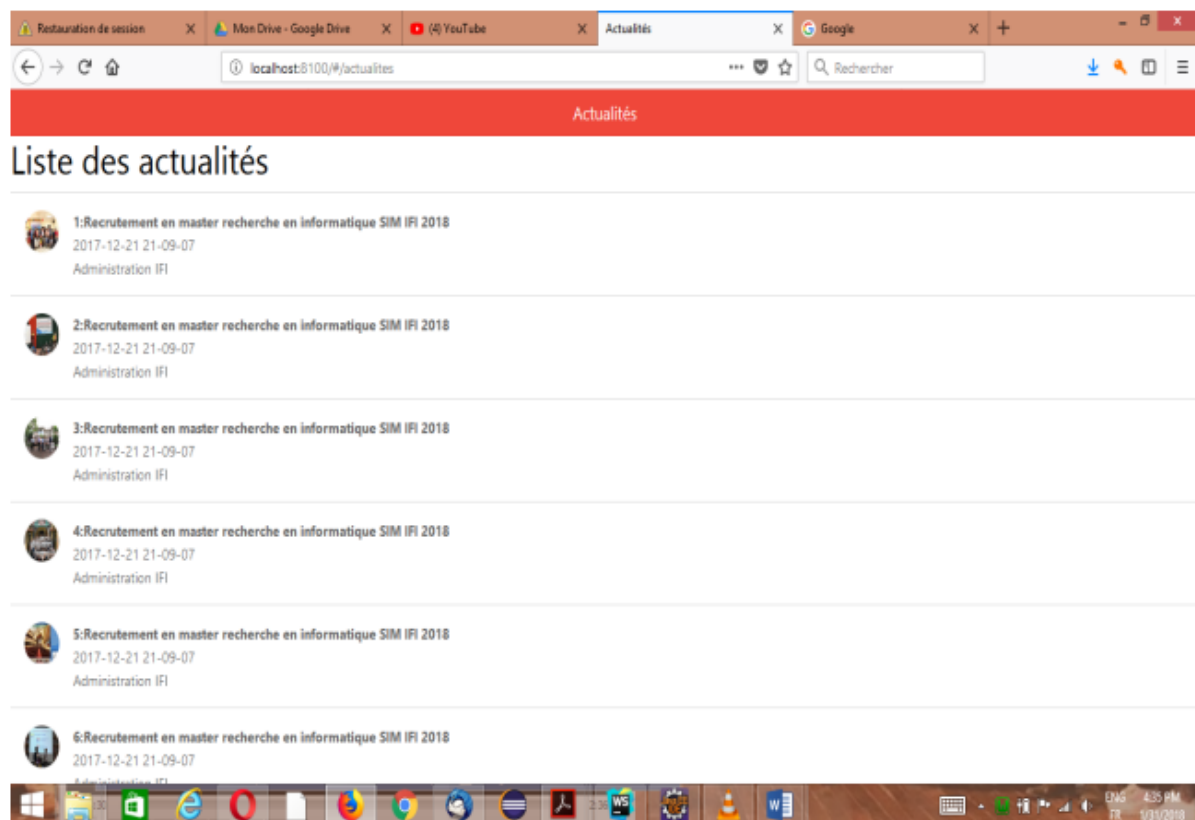


FIGURE 15 – Page d'accueil de la liste des actualités

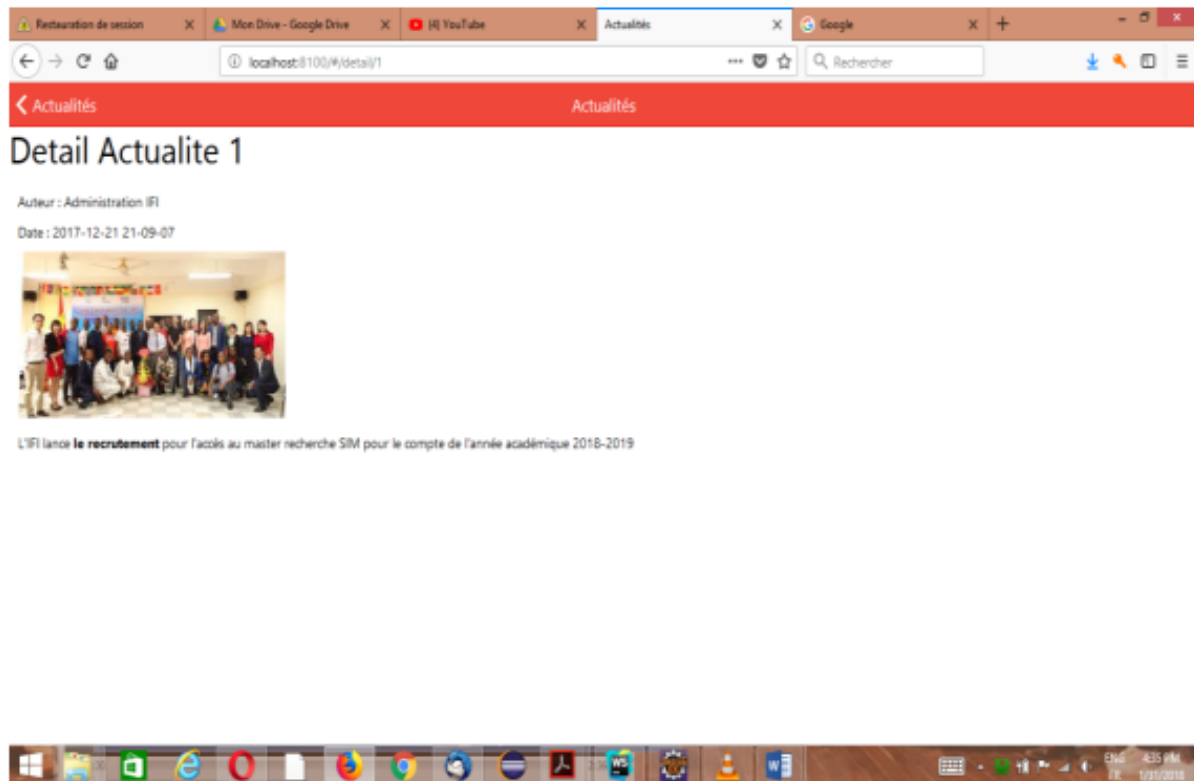


FIGURE 16 – Affichage du contenu d'une actualité

l'interface d'accueil de la plate-forme.

Après avoir créé son compte utilisateur, la création de l'application peut être resumée en ces différents points : é

- (a) Cliquer sur le menu Create App : la page suivante nous permet de renseigner le nom de notre application ainsi que le package dans lequel nous souhaiterons la placer. Dans le cas présent, nous faisons la demo tout en mettant en place une application mobile au compte de l'Institut Francophone International IFI et une autre pour l'Université National du Vietnam VNU. Ces applications nous permettront d'accéder au site web de l'IFI et d'autre part VNU avec tout le contenu dont ils disposent.
- (b) L'étape suivante consiste à définir les paramètres généraux de l'application. Il s'agit essentiellement de donner le logo, l'image de notification en cas de nouvelles informations ou d'une mise à jour sur le site, et de définir l'image de background. La figure ci-dessous nous présente ces différents points.
- (c) L'étape suivante est la plus importante car elle regroupe la liste des modules que propose la plate-forme. A disposition, on retrouve des flux rss, des pages web, différents service web des vidéos youtube et bien d'autres modules. Il est bien possible de regrouper plusieurs modules pour en faire ce que le souhaite avoir dans notre application. Plus précisément, nous choisissons le module Web car notre objectif est d'accéder uniquement au site web de VNU. L'interface des modules se présente comme illustrée dans la figure ci-dessous.

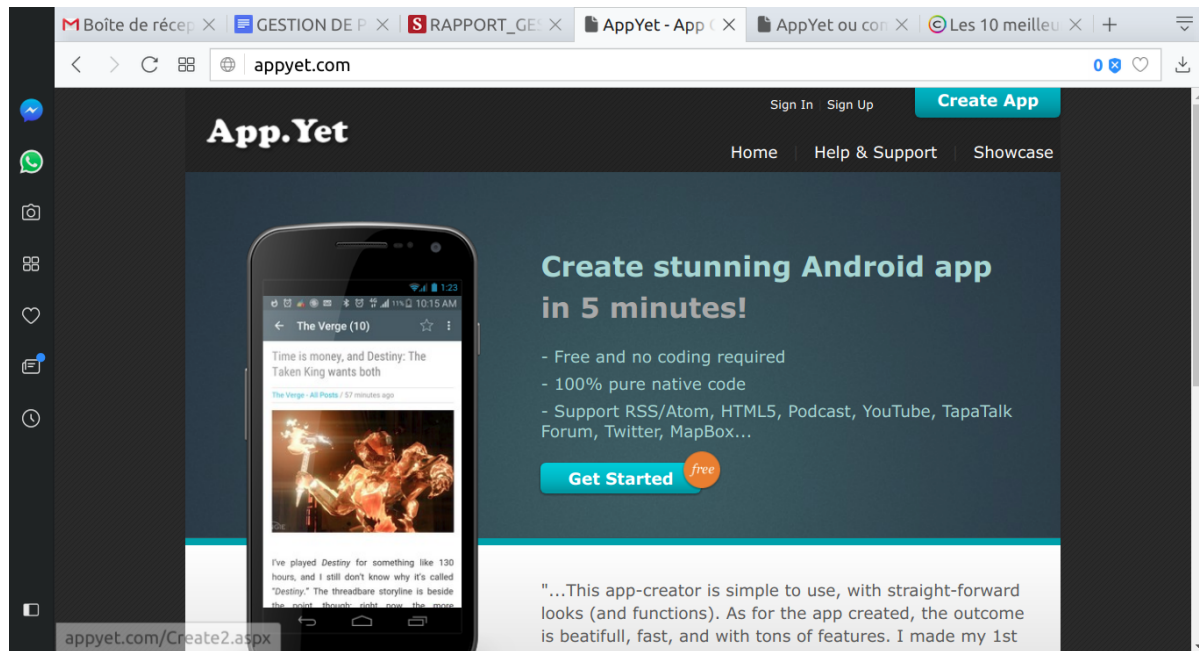


FIGURE 17 – Interface d'accueil de Appyet

AppYet V3

Back to My App

Create New AppYet V3 Android Application

Application Name:

Name for your application

Package Name:

The package name may contain uppercase or lowercase letters ('A' through 'Z'), numbers, and underscores ('_'). However, individual package name parts may only start with letters. For example, applications published by Google could have names in the form com.google.app.application_name

Template: Blank Template

Create application base on predefined template or your existing application. Template is optional.

Create App

Tidal is still great, except for all the nonsense

1 hour ago

Buick and GMC join GM's list of CarPlay and Android Auto brands

1 hour ago

FIGURE 18 – Première étape (définition du nom et du package)

General


Changes has been saved succesfully! Please [Submit to Build](#) then reinstall new apk to see your changes on Android.


Package Name: com.vnu


Application Name:
[Localize...](#)
Name for your application

Version Name:
Version name for your application

Version Code: 2
Version code is unique and automatically generated

Application Icon: 
[Change icon](#) | [Reset to default](#)

Notification Icon: 
[Change icon](#) | [Reset to default](#)

Header Image: 
[Change image](#) | [Reset to default](#)

Content Rating:
Rate your application content. Use this [Google Play content rating guideline](#)

Build Status: Pending

[Save Changes](#)

[Delete this application](#)

FIGURE 19 – Première étape (définition du nom et du package)

Modules

[Go to New Module](#)

	Module Type	Name	Sort Order▲	Enabled	
🕒	FeedQuery	[last_24hours]	2	<input checked="" type="checkbox"/>	Details
🔔	FeedQuery	[all_unreads]	4	<input checked="" type="checkbox"/>	Details
★	FeedQuery	[all_stars]	6	<input checked="" type="checkbox"/>	Details
🎵	FeedQuery	[all_podcasts]	8	<input checked="" type="checkbox"/>	Details
...	GroupDivider	GroupDivider1	10	<input checked="" type="checkbox"/>	Details
🎯	Placeholder	FeedPlaceholder	12	<input checked="" type="checkbox"/>	Details
🎯	Placeholder	ForumPlaceholder	34	<input checked="" type="checkbox"/>	Details
...	GroupDivider	GroupDivider2	1000	<input checked="" type="checkbox"/>	Details
📁	Downloads	[downloads]	1002	<input checked="" type="checkbox"/>	Details
🔄	Sync	[sync_now]	1004	<input checked="" type="checkbox"/>	Details
📁	Themes	[switch_theme]	1006	<input checked="" type="checkbox"/>	Details
⚙️	Settings	[settings]	1008	<input checked="" type="checkbox"/>	Details
🔍	Explore	[explore]	1010	<input checked="" type="checkbox"/>	Details

Show Page: 1 (Total Records: 13)
Records Per Page: ▼

New Module

Feed (RSS/Atom)
Feed Query
Web
TapaTalk Forum
Twitter (Beta)
Post (Beta)

Feed: Blogger
Feed: Flickr
Feed: Pinterest
Feed: Tumblr
Feed: Wordpress

Group Divider

FIGURE 20 – Liste des modules proposés par Appyet)

FIGURE 21 – Interface pour la création du module)

FIGURE 22 – Interface pour la création du module)

(d) Création du module

La création du module intervient après que nous aillons fait le choix du type d module que nous désirons mettre en place. Nous précisons à ce niveau le nom et le numéro d'ordre du menu. Le numéro d'ordre est important dans le cas où nous souhaitons construire notre application à partir de plusieurs modules. Nous donnons VNU INFOS comme nom de notre module et au premier ordre vu que nous n'utiliserons qu'un seul module.

(e) Configuration ou paramétrage du module

La configuration du module consiste à définir le type c'est-à-dire s'il s'agit d'un lien url ou bien un contenu html ainsi que de la façon dont nous souhaitons accéder au site (browser). En utilisant le lien web, cela suppose que nous partons déjà d'un site existant. Dans le cas contraire, nous devons écrire le code html de notre site web qui sera directement compilé. Nous utilisons le lien du site web de VNU qui est `https://vnu.edu.vn/eng/`.

(f) Compilation de l'application

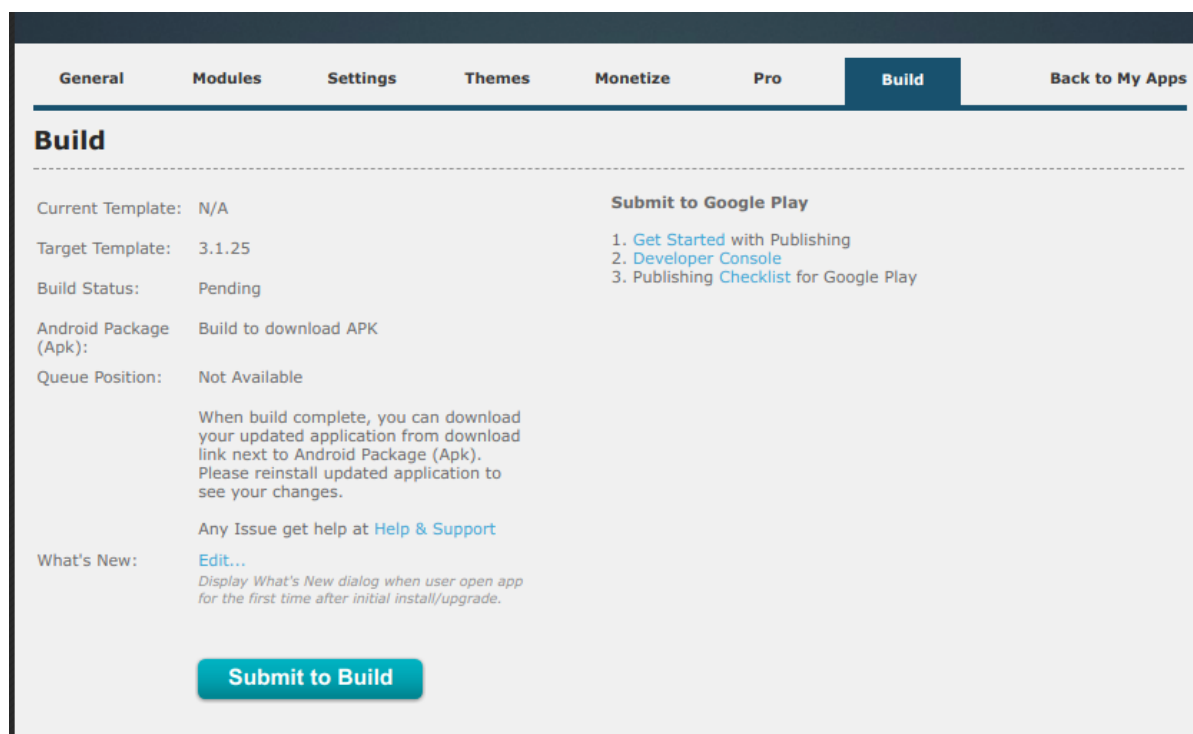


FIGURE 23 – Interface pour la création du module)

Cette partie pourrait se résumer comme état la dernière partie de la construction de notre application. Après avoir passé les étapes précédentes, il ne nous reste plus qu'à compiler notre application qui sera prête à mettre en production.

Pour compiler, il suffit de cliquer sur Submit to Build pour générer l'exécutable de l'application. Une fois le build terminé, un lien de téléchargement sera envoyer par envoyer dans la boîte de messagerie utilisée pour l'ouverture du compte.

L'application peut donc être téléchargée et installer sur tout appareil Android. Après installation, notre application les différentes interfaces de notre application sont présentées à partir des images ci-dessous.

Comme toute technologie, Appyet présente ses forces et ses faiblesses qui peuvent être regroupées comme suit :

1. Les avantages

- **Le premier avantage de Appyet est son accès libre** : Il n'est pas nécessaire de disposer d'une licence pour utiliser les services que propose la plateforme.
- **La simplicité** : les différentes étapes de création de l'application sont très simple et facile à comprendre. Application apparaîtra pour certains un grand mot mais Appyet permet d'en a procurer le plus simplement possible tout en produisant un contenu sympathique et utile.
- **L'existence des modules prédéfinis** : Appyet regorge un bon nombre de modules prédéfinis qui peuvent être combinés pour produire une application avec beaucoup de fonctionnalités.

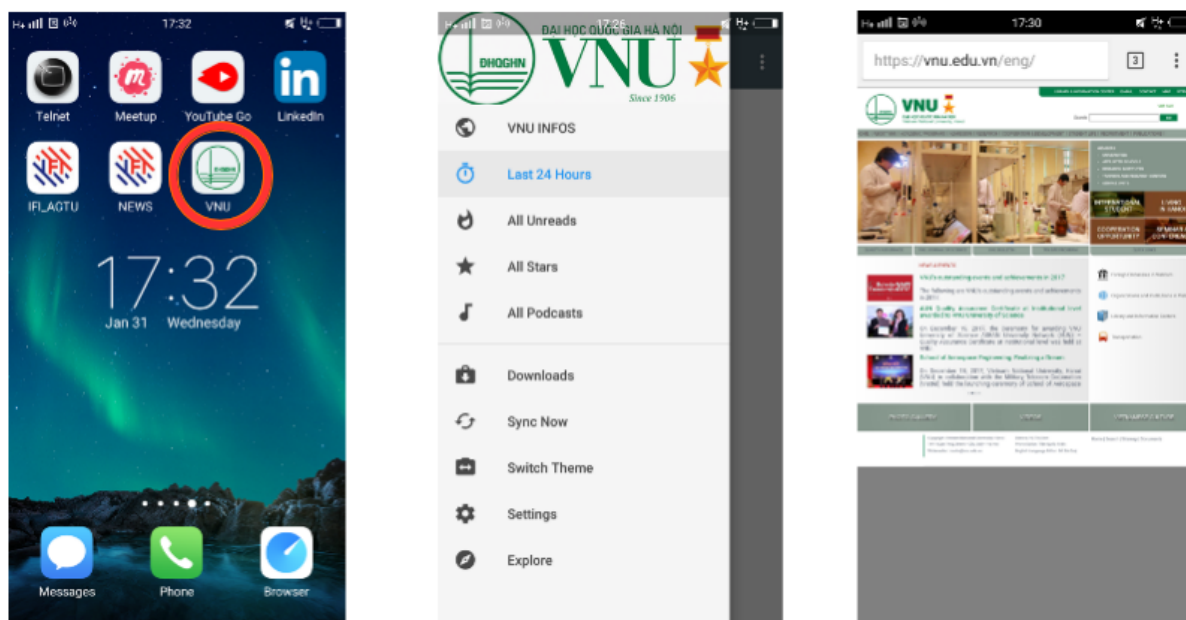


FIGURE 24 – Déroulement de l'application VNU)

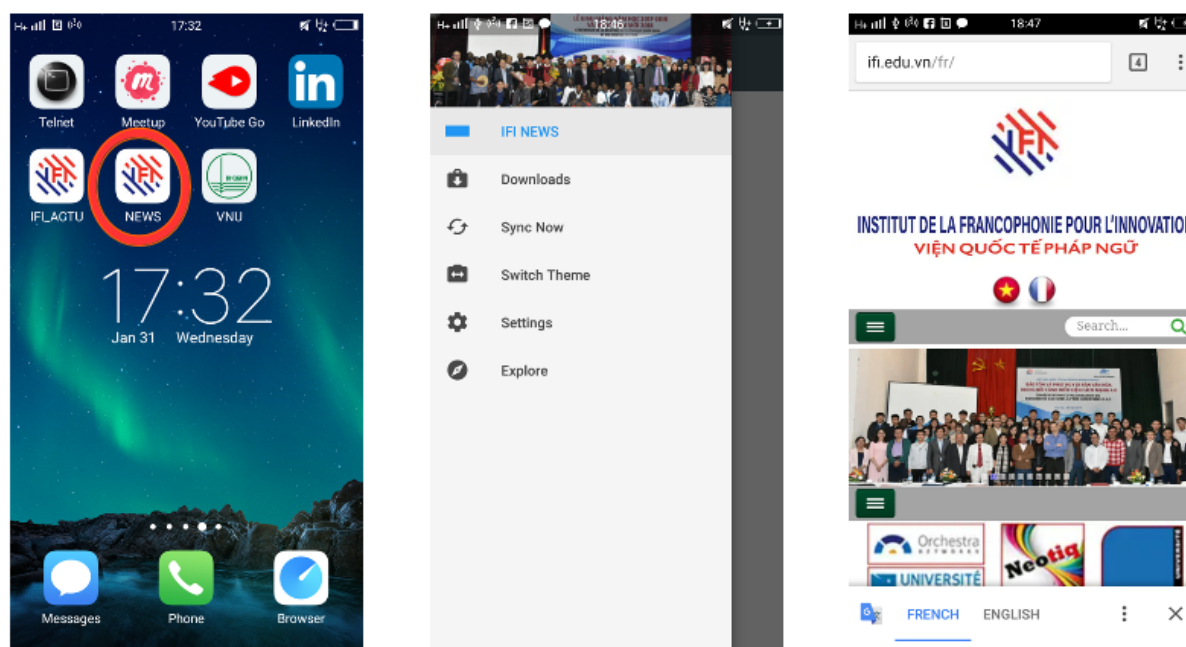


FIGURE 25 – Déroulement de l'application NEWS (site web de IFI))

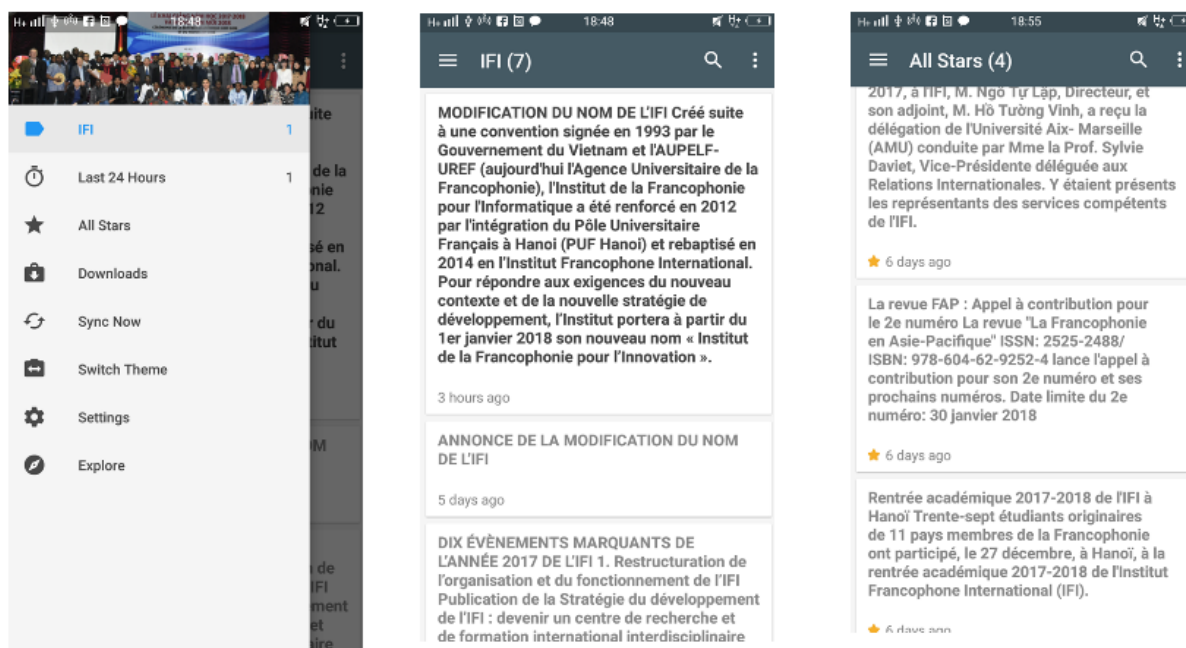


FIGURE 26 – Déroulement de IFI_ACTU (file d'actualité de IFI)

- **Les mises à jours** : Pour mettre à jour l'application, il suffit de télécharger la nouvelle version et procéder à son installation, l'ancienne version est automatiquement remplacée par la nouvelle version.
- **La mise en vente** : les applications de Appyjet sont commercialisable sur playStore. Pour y accéder, il suffit d'avoir un compte développeur soit de payer à la société Appyjet qui se chargera de la publication.

2. Les inconvénients de l'outil Appyjet

Les limites de la plate-forme peuvent être résumées comme suit :

- **Les modules sont prédéfinis et ne peuvent être modifiés par l'utilisateur** : le fait que les modules sont prédéfinis limitent la flexibilité et les besoins de l'utilisateur du fait qu'il s'avère limité en terme de personnalisation.
- Les applications de Appyjet sont uniquement dédiées à des plate-formes Androïde. L'exécutable est uniquement d'une extension .apk.

Première partie

Conclusion Générale et Perspectives

1. Bilan

Dans le cadre du cours de Gestion de projet, nous avons réalisé un système androïde de file d'actualités dont le but est à la fois mettre sur la toile toutes les informations de l'IFI mais et surtout faire suivre les internautes les nouvelles actualités en temps réel. Cette approche a été expérimentée sur deux plateformes à savoir l'université nationale de Vietnam (VNU) et celle de l'institut francophone international.

Notre application a la capacité d'être utilisable sur les téléphones Androïdes, permettre aux utilisateurs d'ajouter les favoris sur les actualités de son gré, Ensuite elle ouvre deux volets d'application sur les deux plate-formes d'expérimentations à savoir : les parties actualités et la partie du site web en tant que tel.

Ce projet nous a permis de mettre en pratique les notions acquises dans le module de gestion de projet, mais aussi de consolider nos compétences en programmation d'applications Web et Androïde en utilisant les technologies telles que Angular 4, IONIC, NodeJS, App.yet, Java qui sont aujourd'hui les outils les plus usités. Ce système manque malheureusement quelques fonctionnalités telles que la possibilité de son usage avec les terminaux tels que Ipad, Iphone d'ou la limite du framework open-source utilisé qui ne génère pas l'exécutable compatible a ces terminaux.

2. Perspectives

Nous envisageons de poursuivre notre travail par une souscription à la licence du framework App.yet, ainsi pour créer les fichiers exécutables à tout type de téléphones pour permettre à l'application d'avoir un champs utilisation assez grand et plus accessible.

1 Difficultés

Références

Références

- [1] webstorm. The smartest javascript ide. [https ://www.jetbrains.com/webstorm/](https://www.jetbrains.com/webstorm/), consultée le 11 novembre 2017.
- [2] android. The official ide for android. [https ://developer.android.com/studio/index.html](https://developer.android.com/studio/index.html), consultée le 08 novembre 2017.
- [3] appyet. Create stunning android app. [http ://appyet.com/](http://appyet.com/), Le lundi 05 septembre 2016, consultée le 20 Janvier 2018.