PROGRAMMATION EN JAVA

**PROJET – RAPPORT 3**

Nom Du Projet

**Bon-Esprit**

Nombre de la group: **03**

Membres:

1) DANG DIEM LINH 18126023

2) LUONG LONG HA 18126014

3) DUONG QUANG VINH 18126038

**Tableau de contenu**

[**L’HISTOIRE DE VERSIONS** 3](#_Toc68302579)

[**INTRODUCTION** 4](#_Toc68302580)

[**ANALYSE ET CONCEPTION** 6](#_Toc68302581)

[**IMPLÉMENTATION** 6](#_Toc68302582)

[**EXEMPLES DE DONNÉES** 15](#_Toc68302583)

[**RÉSULTAT** 21](#_Toc68302584)

[**PLAN** **Error! Bookmark not defined.**](#_Toc68302585)

[**RÉFÉRENCES** 22](#_Toc68302586)

# **L’HISTOIRE DE VERSIONS**

|  |  |  |  |
| --- | --- | --- | --- |
| N.o | Version | La date de création | Description |
| 1 | **Bon-Esprit 01** | 10/3/2021 | Le code qui implémente les classes et les associations entre les classes dans l'application. Cette version est **une démo incomplète** du comportement des classes. |
| 2 | **Bon-Esprit 02** | 3/4/2021 | + **Base de données** supplémentaire pour l'application: les classes commencent à se lier avec la base de données.  + Ajuster et ajouter un certain nombre de méthodes des classes. |
| 3 | **Bon-Esprit 03** | 2/5/2021 | Déployer du code sur **l'interface à l'aide de Swing.** Cette version se traduira par une application relativement complète qui peut être utilisée au niveau de l'interface utilisateur graphique. |

# **INTRODUCTION**

1. **LE BUT DU RAPPORT**

Pour développer une application idéale, outre le backend, le **frontend** est également très important. Frontend décide de la capacité maximale des utilisateurs à exploiter l'application, leur offrant ainsi l'expérience d'application la plus intuitive et la plus spécifique.

Dans les rapports précédents, comment organiser la classe ainsi que comment configurer une base de données pour l'application. Ce rapport se concentrera sur la façon dont **l'interface** est conçue et mise en œuvre pour BonEsprit à l'aide de **Java Swing**.

1. **QUELLES SONT LES PRINCIPALES CHARACTERISTIQUES DE L'INTERFACE DE BONESPRIT?**
2. **Qu’est-ce que BonEsprit?**

**BonEsprit** est une application qui interroge des informations simples liées à des questions **psychologiques,** donc certains des contenus indispensables de l'application sont**:**

* **Articles** d'information sur des problèmes psychologiques spécifiques**.**
* Informations **post-thérapeute.**

En outre, l'application ajoute également des questions de **quiz** pour tester la capacité de l'utilisateur à souffrir **de problèmes psychologiques.**

1. **De quoi a besoin l’interface BonEsprit?**

Sur la base des éléments essentiels de l'application, l'interface idéale de BonEsprit comprendrait:

* L'interface affiche des **informations sur chaque article,** notamment: le titre, la description, les symptômes, le traitement et le lien vers son quiz.
* L'interface d'affichage du **quiz** comprend des questions, chaque question a 4 options.
* L'interface affiche les informations des **thérapeutes.**

Parallèlement à cela, BonEsprit doit également ajouter des interfaces qui aident les utilisateurs à utiliser et à interagir facilement sur l'application, telles que:

* L'interface affiche les informations de **profil.**
* L'interface affiche les informations de **compte.**
* L'interface affiche une **liste des résultats du quiz.**

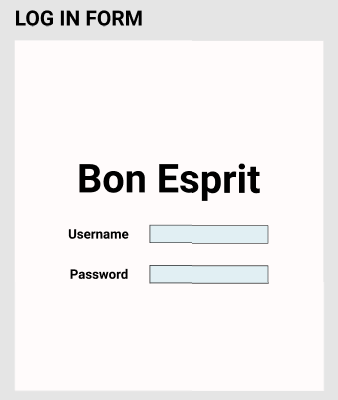
Pour les **utilisateurs experts,** les interfaces supplémentaires sont:

* Interface de **gestion des postes.**
* Interface **ajouter / modifier le message.**
* Interface **ajouter / modifier le quiz.**

# **ANALYSE ET CONCEPTION**

**LES ÉCRANS ATTENDU DANS BONESPRIT**

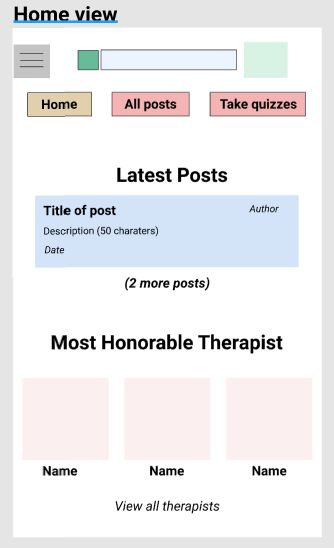
1. **Se Connecter**



Cet écran est pour **se connecter**.

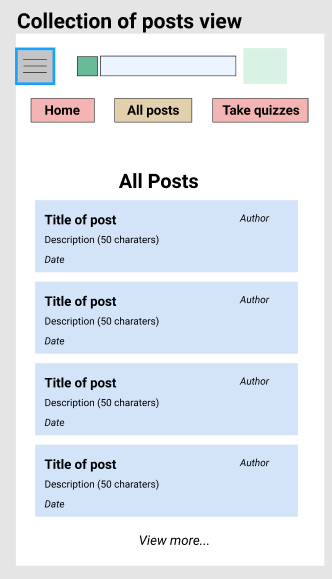
L'utilisateur doit entrer le nom d'utilisateur et le mot de passe pour se connecter à l'application.

1. **Les écrans d’accueil**



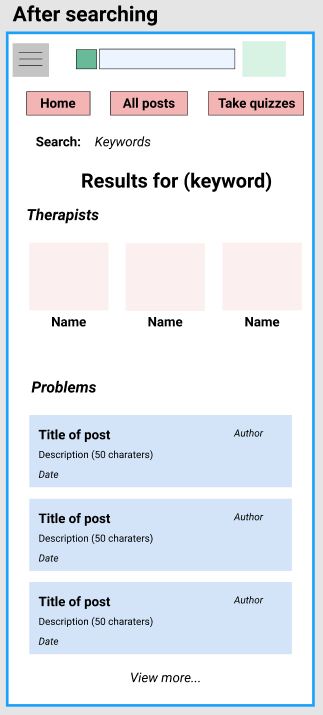
Cet écran présentera **les points forts** de l'application, notamment:

* Les articles les **plus récents**.
* Des **experts de confiance** désormais actifs dans l'application.



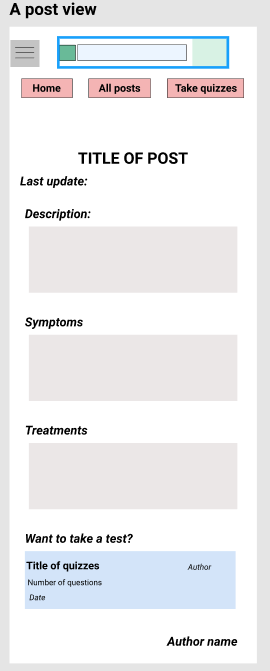
Cet écran affichera **tous les articles** actuellement dans l'application.

Pour éviter que les informations ne soient diluées, seuls **4 articles** sont affichés à la fois, dans l'ordre **du nouveau à l'ancien**.



Cet écran affichera **les articles** **relatives au mot-clé** de recherche, ainsi que **les auteurs** de ces articles.

1. **L’écran d’informations**

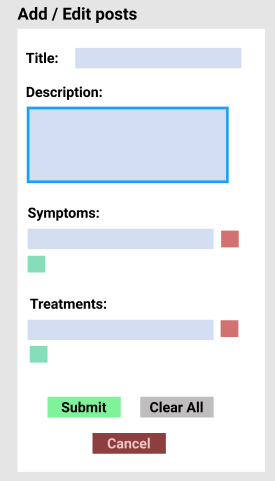


Cet écran affiche des informations sur un article, notamment:

* **Nom** du problème psychologique.
* **Descriptions** de ce problème psychologique.
* **Symptômes** du problème.
* Comment **traiter le problème**.

À la fin, **un quiz** permet de tester la capacité de l'utilisateur à souffrir de problèmes psychologiques. Quelques problèmes **non liées** au quiz maintenance.

1. **L’écran d’ajouter une article**



Il s'agit d'un écran pour les **psychothérapeutes**, utilisé pour **ajouter des problèmes** psychologiques dans l'application.

Chaque fois que vous postez, les experts doivent saisir:

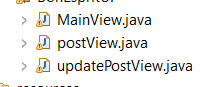
* **Nom** du problème.
* Ajoutez des **symptômes** sous forme de liste.
* Ajoutez les **traitements** sous forme de liste.

Les experts peuvent également **supprimer** les symptômes / traitements tout en ajoutant ou en modifiant des informations.

# **IMPLÉMENTATION**

**Les écrans ci-dessus seront installés en Java à l'aide de Java Swing.**

Tout d’abord, il y a 3 frames principals dans la structure du code

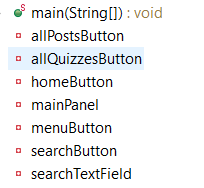


* **MainView** est une frame pour implementer les écrans d’acueil ***(1)***.
* **postView** est une frame pour implementer l’écran d’information ***(2).***
* **updatePostView** est une frame pour implementer l’écran d’ajouter l’article ***(3)***.

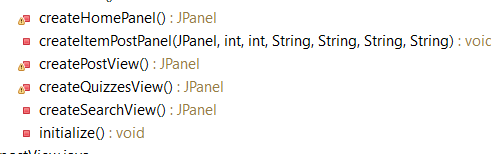
1. **MainView**

Il y a 2 sections, la section **En-tête** et la section **Sous-cadre ci-dessous**, affichées sous la forme d'un **onglet**.

* Dans **l'en-tête**, les propriétés suivantes sont installées:

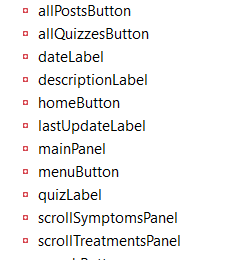
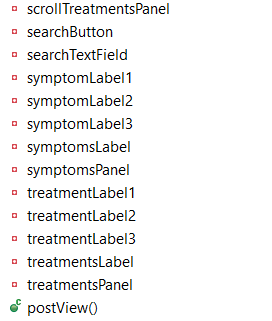


* Dans le cadre secondaire, il y a 4 cadres comprenant home, allPost, allQuizzes et searchPanel qui sont installés comme suit:



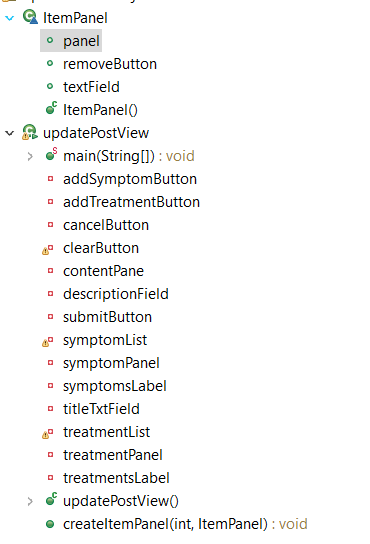
1. **postView**

Pour afficher toutes les informations requises pour un problème, les propriétés suivantes sont définies dans **postView**:



1. **updatePostView**

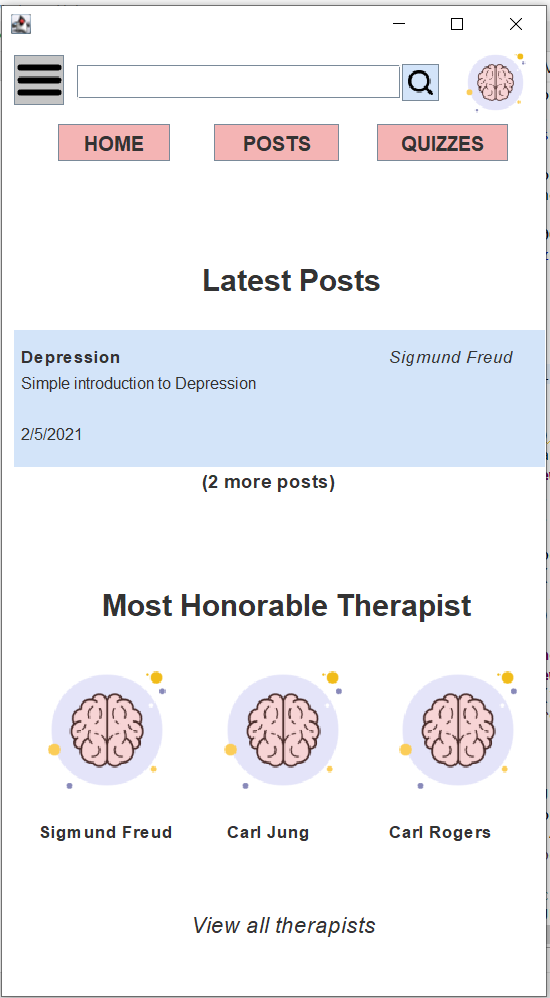
Pour faciliter l'ajout et la modification des informations d'un problème par le thérapeute, **updatePostView** inclut les propriétés suivantes:



# **EXEMPLES DE DONNÉES**

1. **L’écran d’acueil**

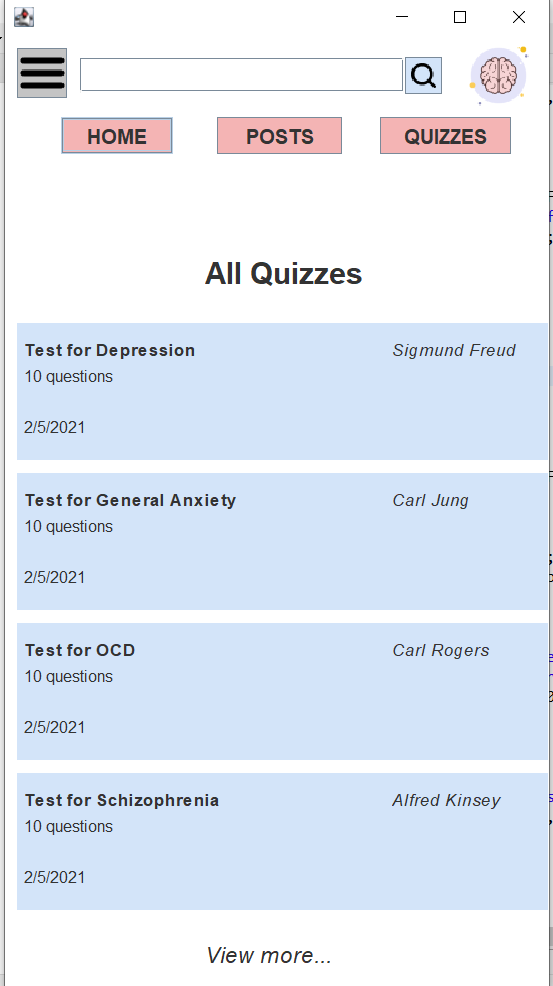
***Home View***



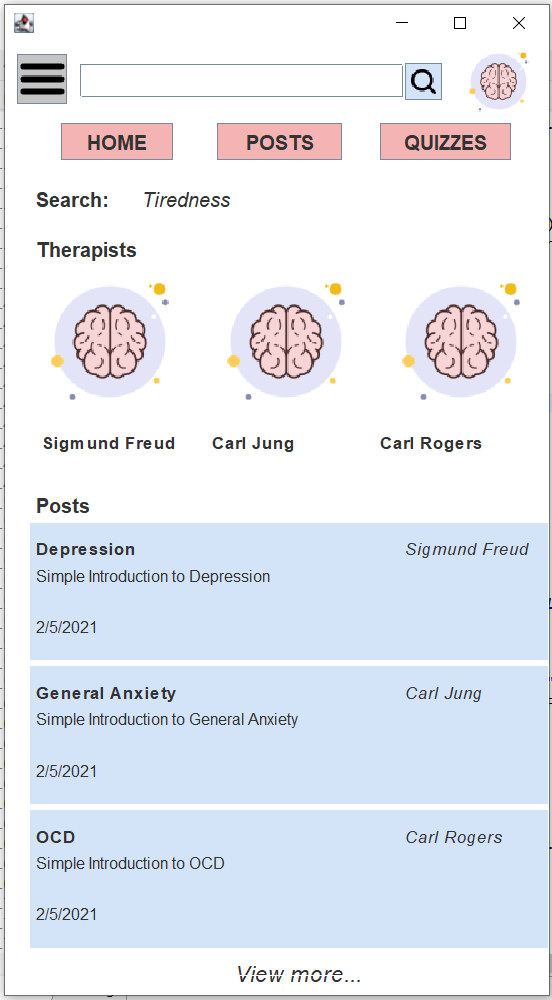
***All Post View***



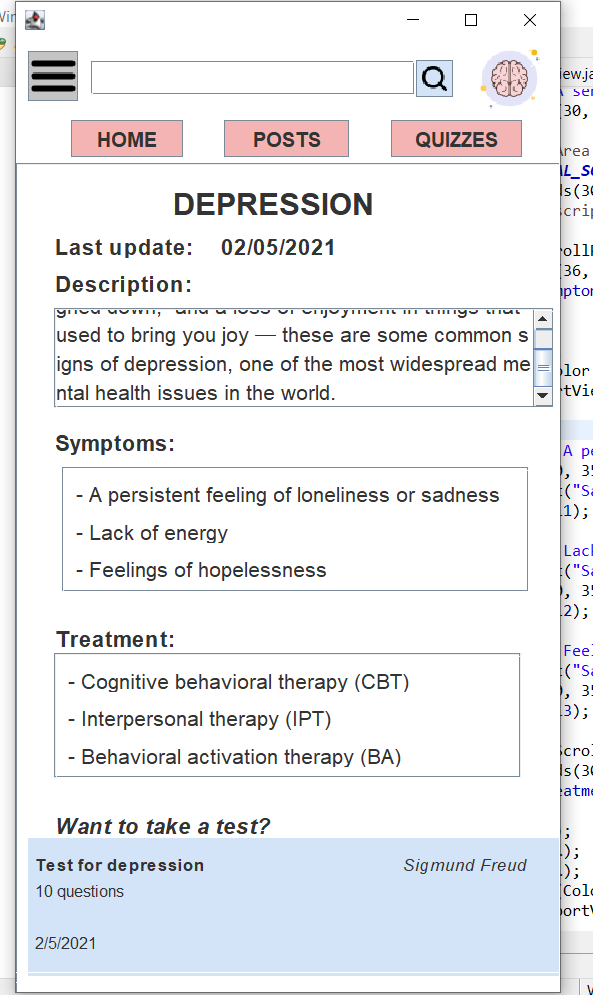
***All Quizzes View***



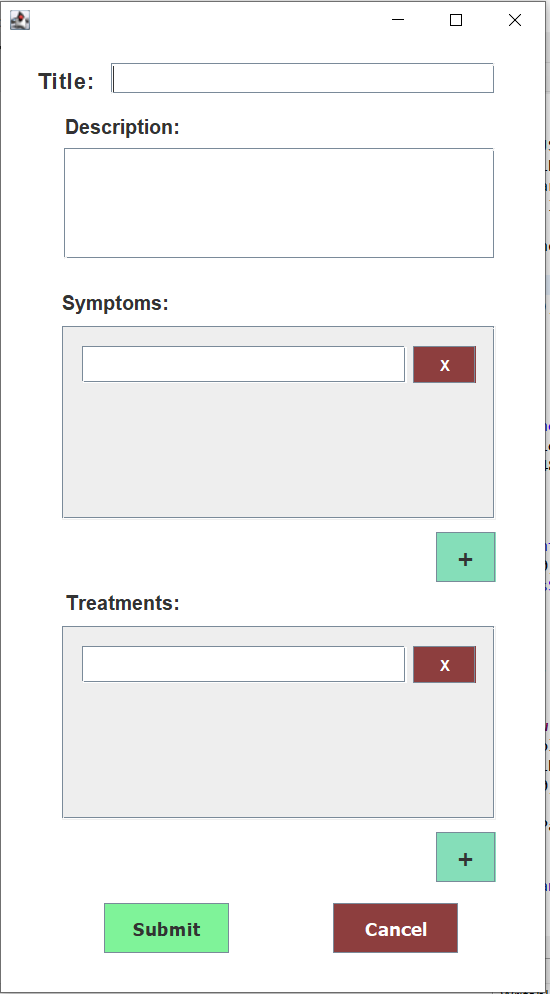
***Searching View***



1. **L’écran d’information d’une article**



1. **L’écran d’ajouter out de modifier une article**



# **RÉSULTAT**

|  |  |
| --- | --- |
| **AVANTAGES** | **DÉSAVANTAGES** |
| Les écrans ont mûri et regorgent de boutons fonctionnels | Besoin d'ajouter des écrans supplémentaires pour basculer entre certaines fonctionnalités spéciales. |
|  | Le problème de la liaison avec la base de données et l'erreur de type de données lors de la consultation n'a pas été complètement résolu. |

# **RÉFÉRENCES**

1. **https://www.w3schools.com/java/**
2. **https://www.w3adda.com/java-swing-tutorial**
3. **https://www.tutorialspoint.com/swing/index.htm**
4. [**https://github.com/ddlinh/bonEsprit**](https://github.com/ddlinh/bonEsprit)**:** La progression du groupe est régulièrement mise à jour sur ce lien.