

# Jeu des capitales

---

Cedric Dos Reis

## Table des matières

Introduction .....	2
Etude d'opportunité .....	2
Analyse fonctionnelle .....	4
Général .....	4
Fonctions globales .....	4
Interfaces .....	5
Fonctionnalités particulières .....	7
Eléments de sécurité .....	7
Analyse Organique .....	7
Généralité .....	7
Description détaillée des méthodes (algorithmes) de résolution .....	7
Variables, Constantes, Record Globales .....	7
Procédure Initialisation .....	8
Procédure ImporterCSVClick .....	9
Fonction ReadDat .....	10
Procédure SelectionReponses .....	10
Procédure VerificationBouton .....	11
Procédure FormCreate .....	11
Procédure BtnNextClick .....	11
Procédure BtnPreviousClick .....	12
Procédure BtnConfirmClick .....	12
Procédure NouveauJeuClick .....	13
Procédure AideUserClick .....	13
Tests .....	15
Plan de test .....	15
Rapport de test .....	16
Conclusion .....	17
Améliorations .....	17
Comparaison énoncé avec projet final .....	17

## Introduction

Dans le cadre de l'atelier TPI, je dois réaliser un jeu concernant la géographie. Ce jeu consiste à trouver la capitale d'un pays. Ce jeu a pour but de mettre à épreuve la connaissance géographique du joueur. Il devra répondre à des questions en choisissant parmi les réponses disponibles.

## Etude d'opportunité

On trouve assez facilement des jeux de capitales sur internet. Le résultat qui apparaît le plus est un quizz nommé « **Quizz : Les capitales du Monde** ». Ce quizz est séparé en 4 parties.



Figure 1: Quizz Les Capitales du Monde

La première partie du jeu est nommée **L'échauffement**. Dans cette manche vous allez devoir répondre à 8 questions sur les capitales des pays et inversement. Vous avez plusieurs choix de réponses, au fil des questions le nombre de réponses possible va augmenter jusqu'à 4. A chaque fois que vous répondez juste, votre nombre de points augmentera. La fin de cette partie se termine après 1 minute que vous ayez fini de répondre aux questions ou pas. Il faut impérativement répondre correctement à au moins 4 questions pour passer à la manche suivante.



Figure 2 : Quizz Les Capitales du Monde - 1ere Manche – L'échauffement

La deuxième partie du jeu est nommé **Le Marathon**. Sur cette manche vous allez devoir répondre à 12 questions. Il est possible de choisir le nombre de réponses possible à chaque question mais cela change le nombre de point obtenue si la réponse est juste selon le nombre de réponse disponible. Vous disposez de 90 secondes pour terminer cette manche. Il faut impérativement répondre correctement à au moins 6 questions pour passer à la manche suivante.

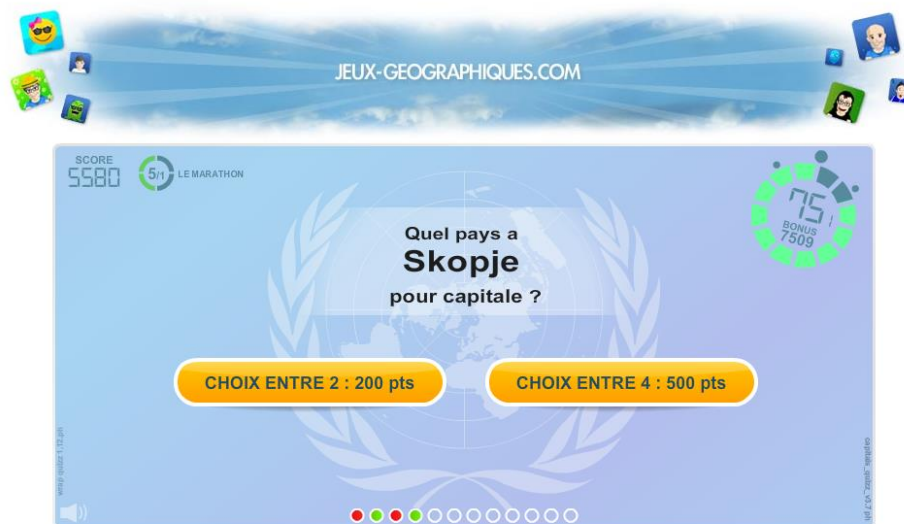


Figure 3 : Quizz Les Capitales du Monde - 2eme Manche – Le Marathon

La troisième partie est nommée **La Course de Cote**. Sur cette manche vous allez devoir répondre à 8 questions. Vous disposez de 7 secondes pour répondre à chaque question. Il faut impérativement répondre correctement à au moins 4 questions pour passer à la manche suivante.

La quatrième et dernière manche est nommée **Le Sprint**. Sur cette manche, il n'y pas un nombre définit de questions. Le but de cette manche est de répondre au maximum de questions en 60 secondes mais si vous répondez faux à une question vous perdrez 5 secondes.

Ce jeu est disponible à cette adresse : <http://www.jeux-geographiques.com/jeux-en-ligne-Quizz-Les-capitales-du-Monde- pageid89.html>

Le jeu développé durant ce cours sera beaucoup plus simple. Il n'y aura pas de limite de temps. Mais elle ne nécessite pas de connexion internet pour pouvoir jouer contrairement au jeu ci-dessus.

## Analyse fonctionnelle

### Général

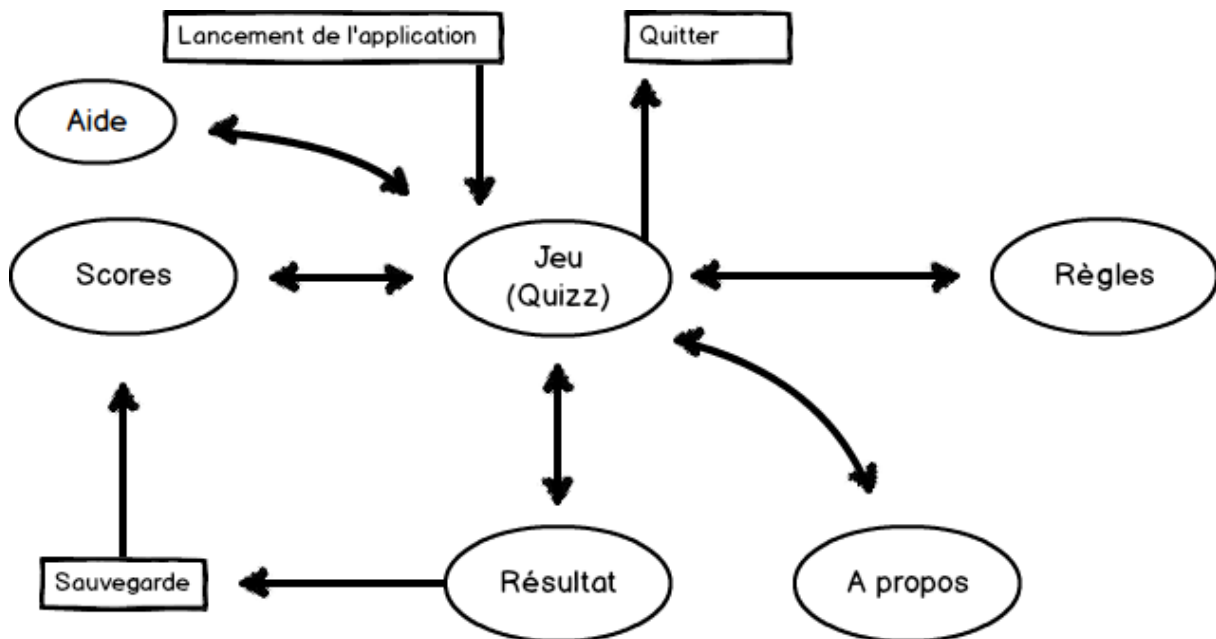


Figure 4 : Schéma global

### Fonctions globales

Le joueur peut commencer/recommencer une partie

Le joueur peut choisir le nombre de questions (entre 10 et 15)

Le joueur peut choisir la réponse selon les propositions affichées (QCM)

Le joueur peut naviguer entre les questions

Le score du joueur est affiché à la fin du quizz

Le joueur peut sauvegarder son score

Le joueur peut afficher tous les scores

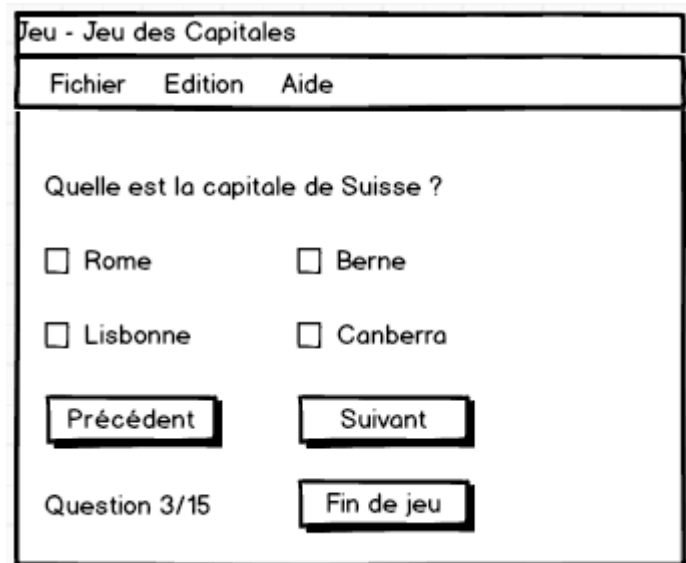
Le joueur peut consulter les règles du jeu

Le joueur peut consulter l'aide de l'application

Le joueur peut quitter le jeu

## Interfaces

La figure 5 nous montre l'interface de la fenêtre de jeu. Au lancement du jeu, c'est la première fenêtre qui apparaît. On peut voir ici la question sur la Suisse et juste en dessous 4 possibilité de réponses dont l'une d'elles est la bonne. Le joueur peut se déplacer entre les questions et mettre fin au jeu à n'importe quel moment.



Jeu - Jeu des Capitales

Fichier Edition Aide

Quelle est la capitale de Suisse ?

☐ Rome ☐ Berne

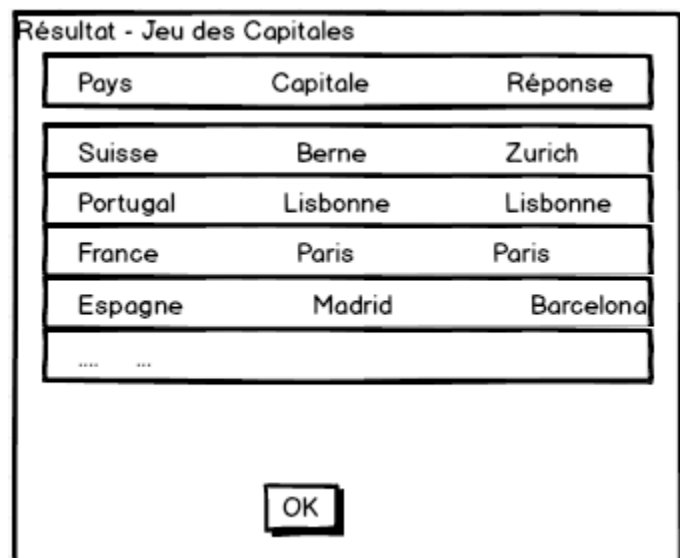
☐ Lisbonne ☐ Canberra

Précédent Suivant

Question 3/15 Fin de jeu

Figure 5 : Interface de la fenêtre de jeu

Lorsque le joueur a fini de répondre au quizz, le résultat des questions est affiché sur une nouvelle fenêtre dont voici l'interface. On peut voir les pays questionnés ainsi que leur capitale et tout à droite se trouvent les réponses du joueur.



Résultat - Jeu des Capitales

Pays	Capitale	Réponse
Suisse	Berne	Zurich
Portugal	Lisbonne	Lisbonne
France	Paris	Paris
Espagne	Madrid	Barcelona

OK

Figure 6 : Interface de la fenêtre des résultats

Le joueur à la possibilité d'afficher tous les scores enregistré depuis le début. Voici l'interface de l'affichage des scores. On peut voir que le nom du joueur est affiché ainsi que le nombre de questions qu'il avait choisi et l'on retrouve tout à droite le score que le joueur a obtenu.

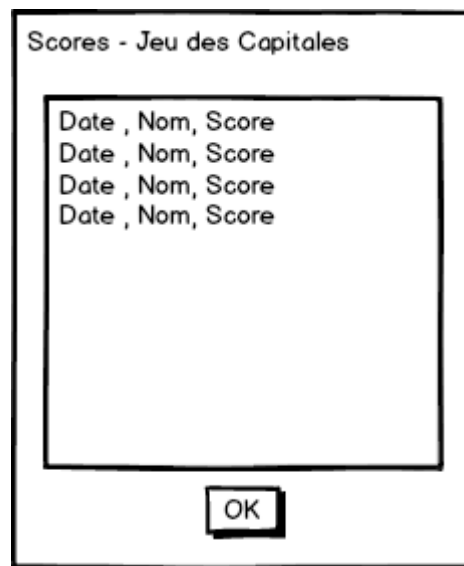


Figure 7 : Interface de la fenêtre des scores

Voici l'interface de la fenêtre à propos. Elle contient quelques informations sur l'application.

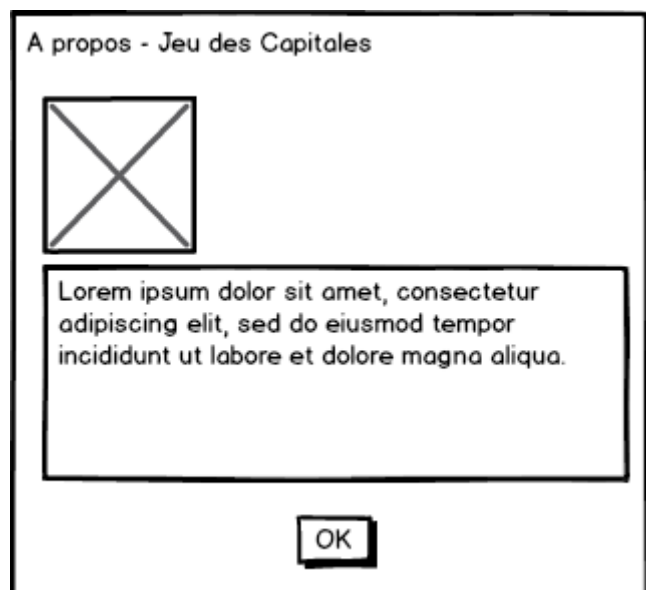


Figure 8 : Interface de la fenêtre A propos.

## Fonctionnalités particulières

L'application lit et écrit sur des fichiers externes de type CSV, DAT, INI et TXT

## Éléments de sécurité

Si un fichier externe est inexistant, l'application le crée automatiquement si c'est possible sinon un message d'erreur informera l'utilisateur qu'un fichier est manquant

## Analyse Organique

### Généralité

Ce projet a été développé grâce au logiciel Delphi 5

### Description détaillée des méthodes (algorithmes) de résolution

#### Variables, Constantes, Record Globales

**TableauQuestions : array [0..15] of Integer;** Ce tableau à une dimension contient dans chacune de ses cases un chiffre qui fait référence à un emplacement dans le fichier data.dat. il permet ainsi de récupérer les infos d'une question (Pays + Capitale + Continent)

**TableauReponses : array [0..15] of String;** Ce tableau à une dimension contient dans chacune de ses cases la réponse que l'utilisateur a sélectionné selon la question

**TableauReponses2: array[0..15,0..3] of String;** Ceci est un tableau à deux dimensions. Les index de la première dimension font référence au numéro de la question. Les cases de la deuxième dimension contiennent les 4 réponses disponibles pour chacune des questions.

**TableauIndex:** Ce tableau à une dimension contient dans chacune de ses cases l'index de la réponse que l'utilisateur a sélectionné selon la question (Elle est utilisée lorsque l'utilisateur revient sur une question, il peut ainsi savoir quelle réponse il avait choisie).

**NbQuestion: Integer ;** Cette variable change de valeur à chaque fois que l'utilisateur passe à une autre question. Elle contient le numéro de la question en cours.

#### Type TCapitale = Record

**Nom : String[35];**

**Pays : String[35];**

**Continent : String[9];**

**end;**

Ce Record est utilisé pour récupérer les informations sur les capitales depuis le fichier data.dat



**Type TScore = Record****Score : integer;****Nom : String[30];****Date : TDate;****NbQuestions : integer;****end;**

Ce Record permet de stocké les informations de la partie avant l'enregistrement du score mais aussi de récupérer les scores du fichier score.dat et les afficher

**DATAWORLD : string= './data.dat';****DATSCORE : string='./Score.dat';**

Les constantes ci-dessus sont en quelque sorte des raccourcis lors de l'utilisation de ces deux fichiers

**Procédure Initialisation**

Cette procédure comme son nom l'indique est l'initialisation de l'application/jeu. Elle va créer les questions qui seront posées tout au long de la partie.

Assigne le fichier data.dat en lecture seule

Récupère la longueur du fichier data.dat

```
//Vérification de la génération pour ne pas avoir de doublon
while j <= NombreTotalQ - 1 do
begin
  nombre := random(FileLength+1); //
  exist := false;
  for i := 0 to j do
  begin
    //Verifie que l'emplacement aléatoire n'est pas déjà dans le tableau des questions
    if TableauQuestions[i] = nombre then
    begin
      exist := true;
      break;
    end;
  end;
  //Si ce n'est pas un doublon, la valeur est ajoutée au tableau des questions
  if not(exist) then
  begin
    TableauQuestions[j] := nombre;
    inc(j);
  end;
end;
```

Tant que **j** est plus petit que le **nombre total de questions**

Initialise un **nombre** random par rapport à la **longueur du fichier** data.dat

Initialise **exist** à faux

Si le nombre random est déjà dans le Tableau alors

Initialise **exist** à vrai

Quitte la boucle

Si **exist** est faux alors

Ajoute le **nombre** random dans le **tableau**

Incrémente **j**

Fin de boucle

Positionne le curseur de lecture dans le fichier data.dat à l'emplacement défini

Récupère les informations (Capitale, Pays) dans le fichier data.dat depuis l'emplacement ci-dessus

Sélectionne des réponses fausses choisies aléatoirement dans le fichier data.dat pour le choix multiple

Ferme le fichier data.dat

Affiche le numéro de la question dans la barre de status

Active le bouton de validation du formulaire

### Procédure ImporterCSVClick

Cette procédure récupère tous les enregistrements dans le fichier capitales.csv pour les transférer dans le fichier data.dat

Si le fichier capitales.csv existe alors

Assigne le fichier capitales.csv en lecture seule

Assigne le fichier data.dat en lecture seule si il existe et en écriture si il n'existe pas

Tant que ce n'est pas la fin du fichier capitales.csv alors

Lit la première ligne du fichier capitales.csv

Récupère les valeurs (Capitale, Pays) et les ajoute à une variable de type **TCapitale**

Supprime les valeurs récupérées

Ecrit dans le fichier data.dat les valeurs qui sont dans la variable de type **TCapitale**

Libère le fichier capitales.csv

Libère le fichier data.dat

Fin de boucle

Si le fichier Capitales.csv alors

Affiche un message d'erreur

### Fonction ReadDat

Cette fonction permet de lire les enregistrements dans le fichier data.dat. Elle reçoit en paramètre un intègre qui correspond à un emplacement dans le fichier data.dat. Elle retourne un record de type **TCapitale**

Assigne le fichier data.dat en lecture seule

Positionne le curseur de lecture à l'emplacement reçu en paramètre

Lis l'enregistrement à l'emplacement défini ci-dessus et récupère les informations (Capitale, Pays)

Libère le fichier data.dat

Retourne les informations récupérées ci-dessus

### Procédure SelectionReponses

Cette procédure va choisir au hasard 3 réponses en plus de la vraie réponse pour la question en paramètre. Elle reçoit en paramètre l'emplacement de la question dans le fichier data.dat

Sélectionne un **nombre** random entre 0 et 3

Assigne le fichier data.dat en lecture seule

Boucle for à 4 passages

    Récupère une capitale au hasard dans le fichier data.dat

    Ajoute la capitale à la liste des réponses disponible pour la question

Fin de boucle

Libère le fichier data.dat

Récupère la vraie réponse à la question reçue en paramètre

Ajoute la vraie réponse à la liste des réponses disponible. Elle remplace une des fausses réponses. La réponse est ajoutée à la liste des réponses à l'emplacement défini par le **nombre** random ci-dessus.

**Procédure VerificationBouton**

Cette procédure permet d'activer /désactiver certains bouton selon des conditions

Si la question actuelle est égale au nombre total de question alors

Désactive le Bouton Suivant

Sinon

Active le bouton Suivant

Si la question actuelle est plus grande que 1 alors

Active le bouton Précédent

Sinon

Désactive le bouton Précédent

**Procédure FormCreate**

Cette procédure se lance au lancement de l'appliaction. Elle permet de lancer d'autres procédures pour le lancement du jeu.

Crée le fichier NbQuestion.ini

Lance la procédure de randomisation

Si le fichier NbQuestion.ini existe alors

Récupère la valeur contenu dans le fichier NbQuestion.ini

Sinon

Affiche un message d'erreur expliquant à l'utilisateur comment créer le fichier

**Procédure BtnNextClick**

Cette procédure permet à l'utilisateur de passer à la question suivante. Elle sauvegarde dans un tableau les réponses disponible ainsi que la réponse de l'utilisateur.

Boucle (for **i** := 0 to 3) à 4 passage

Récupère la réponse en **i** pour la sauvegarder dans un tableau à deux dimensions (**TableauReponses2**)

Fin de boucle

Enregistre l'index de la réponse que l'utilisateur à sélectionné dans le tableau **TableauIndex**

Enregistre le nom de la capitale qui correspond à la réponse de l'utilisateur le tableau **TableauReponses**

Incrémente la Question

Boucle (for **i** := 0 to 3) à 4 passage

Récupère la réponse en **i** pour la prochaine question (Est vide si l'utilisateur arrive à cette question pour la première fois)

Fin de boucle

Sélectionne la réponse que l'utilisateur avait sélectionnée (Est vide si l'utilisateur arrive à cette question pour la première fois)

Récupère des questions pour la question suivante (Utilise la procédure **SelectionneReponses**)

Affiche le numéro de la question dans la barre de statut

Récupère la question selon le numéro de la question en cours

Affiche la question

#### Procédure **BtnPreviousClick**

Cette procédure permet à l'utilisateur de revenir à la question précédente. Elle sauvegarde dans un tableau les réponses disponible ainsi que la réponse de l'utilisateur.

Elle contient exactement le même code que la procédure **BtnNextClick** à une ligne près -> dès incréméte la question

#### Procédure **BtnConfirmClick**

Cette procédure permet à l'utilisateur de valider terminer le jeu et ainsi valider ses réponses.

Enregistre la dernière réponse de l'utilisateur à l'aide de la procédure **BtnNextClick**

Nettoie les ListBox dans la Form **FrmSaveScore**

Initialise **score** à 0

Boucle (for **i**:=0 to NombreTotalQuestion)

Récupère l'enregistrement de la question **i**

Affiche les valeurs dans les ListBox dans la Form **FrmSaveScore**

Si la réponse à la question correspond la réponse de l'utilisateur (**TableauReponses**) alors

Incrément le **score**

Fin de boucle

Affiche le **score**

Si le résultat de le form **FrmSaveScore** est **mrOK** alors

Enregistre dans une variable de type TScore les infos (Score, Nom, Nombre de questions, Date) qui seront enregistrées dans le fichier score.dat

Assigne le fichier score.dat

Si le fichier score.dat n'existe pas alors

Création du fichier score.dat

Sinon

Positionne le curseur à la fin du fichier score.dat

Ecrit dans le fichier score.dat les valeurs qui sont dans la variable de type **TScore**

Libère le fichier score.dat

Fin de boucle

### Procédure NouveauJeuClick

Cette procédure permet de choisir le nombre de question entre 10 et 15. Le jeu se lance automatiquement dès que le choix est fait.

Crée le fichier NbQuestions.ini

Si le résultat de la form **FrmNbQ** = **mrOK** alors

Si 10 est coché alors

**Nombre de question** = 10

Si 15 est coché alors

**Nombre de questions** = 15

Ecrit dans le fichier NbQuestions.ini la valeur de **Nombre de questions**

Libère le fichier ini

Lance le jeu (Lance la procédure **Initialisation**)

### Procédure AideUserClick

Assigne le fichier Aide.txt en lecture seule

Vide le **memo** de la form **F\_Aide**

Tant que ce n'est pas la fin du fichier aide.txt alors

Lis la ligne (**readln**)

Ajoute la ligne au **memo** de la form **F\_Aide**

Fin de boucle

Libère le fichier aide.txt

Affiche la form **F\_Aide**

## Tests

### Plan de test

#### Plan de test

N° Test	Auteur	Scénario	Résultat attendu
T0001	Cedric Dos Reis	le fichier capitales.csv n'existe pas (lors de l'importation)	L'application affiche un message d'erreur
T0002	Cedric Dos Reis	le fichier data.dat n'existe pas	il est créé automatiquement à l'aide du fichier capitales.csv
T0003	Cedric Dos Reis	Les question sont toutes différentes	Il n'a y pas plusieurs fois la même question
T0004	Cedric Dos Reis	La réponse à la question est parmi la liste des réponses disponible	La réponse à la question est dans la liste
T0005	Cedric Dos Reis	Les fenêtres ont une taille fixe	Les fenêtres ne peuvent pas être redimensionnées
T0006	Cedric Dos Reis	Toutes les réponses disponible sont différentes	Il n'y a pas deux fois la même réponse
T0007	Cedric Dos Reis	Il faut entrer une nom pour pouvoir enregistrer le résultat	Bouton d'enregistrement désactivé
T0008	Cedric Dos Reis	Affichage des scores	Les scores sont triés par nombre de questions
T0009	Cedric Dos Reis	Ouverture des autres fenêtres	Elles s'ouvrent au milieu de l'écran et affiche le contenu demandé
T0010	Cedric Dos Reis	Lorsque le joueur a enregistré son score	Un nouveau jeu se lance

Figure 9 : Plan de test pour l'application Jeu des Capitales



**Rapport de test****Rapport de test**

Date	Testeur	N° Test	Résultat obtenu	OK/KO	Fiche Anomalie
26.03.2015	Cedric Dos Reis	T0001	L'application affiche une message d'erreur	OK	FA0001
26.03.2015	Cedric Dos Reis	T0002	L'appilcation crée automatiquement le fichier a partir de capitales.csv	OK	FA0002
26.03.2015	Cedric Dos Reis	T0003	il n'y a pas plusieurs fois la même questions	OK	FA0003
26.03.2015	Cedric Dos Reis	T0004	La réponses à la question est dans la liste	OK	FA0004
26.03.2015	Cedric Dos Reis	T0005	les fenetre ne peuvent pas être redimensionnées	OK	FA0005
26.03.2015	Cedric Dos Reis	T0006	Il n'y a pas deux fois la même réponse	OK	FA0006
26.03.2015	Cedric Dos Reis	T0007	Bouton d'enregistrement désactivé	OK	FA0007
26.03.2015	Cedric Dos Reis	T0008	Les scores sont triés par nombre de questions	OK	FA0008
26.03.2015	Cedric Dos Reis	T0009	Affiche les fenêtres au milieu de l'écran, affiche le contenu demandé	OK	FA0009
26.03.2015	Cedric Dos Reis	T0010	Ferme la fenêtre des scores	KO	FA0010

**Figure 10 : Rapport de test pour l'application Jeu des Capitales**

## Conclusion

### Améliorations

Bien que ce projet soit déjà bien avancé, il n'y aura rien de mal à ajouter des fonctions supplémentaires pour une meilleure utilisation, en voici quelques une :

Possibilité de choisir des questions sur les capitales des différents continents du monde ou de toutes les capitales du monde

Intégrer une petite difficulté au jeu en ajoutant une limite de temps pour terminer le questionnaire.

### Comparaison énoncé avec projet final

Le projet développé durant ce cours correspond largement à l'énoncé rédigé avant le début du développement.

Et il contient tous les éléments demandé.