

Turing Design d'un Jeu

[Vue d'Ensemble](#)

[Le Jeu](#)

[Contrôles](#)

[Les Buts](#)

[Environment](#)

[Structure](#)

[Avatar](#)

[Environment](#)

[Niveau](#)

[Layers](#)

[Floor](#)

[Walls](#)

[furniture](#)

[Decoration](#)

[pickable](#)

[interacted](#)

[interactable](#)

[Groups](#)

[monsters](#)

[triggers](#)

[Interface Utilisateur](#)

[HUD](#)

[Audio](#)

[Index](#)

[Language reference](#)

[Classes tutorial](#)

[Resources](#)

[Extrait de Code Source](#)

[Loading multiple Pic](#)

[Images](#)

[Image Slicing](#)

[Image conversion](#)

[Converting .png into .bmp](#)

Vue d'Ensemble

Donjon est un jeu où le joueur doit explorer un donjon plein de trésors et de périls. Le joueur doit survivre jusqu'à ce qu'il ait atteint la sortie. Le pointage obtenu par le joueur déterminera si il/elle a bien joué.

Le Jeu

Contrôles

Le joueur peut déplacer son avatar (personnage du joueur) vers le haut, le bas, la gauche et la droite en utilisant les touches: "W,S,A,D" sur un clavier ou avec les flèches sur la manette faite pour ce jeu. Pour activer des objets ou attaquer des monstres, la touche "Espace" du clavier ou "A" sur la manette peut être utilisée. La touche "Esc", ainsi que le bouton "Menu" sert à ouvrir le menu.

Les Buts

Le but principal est de trouver la sortie sans que les points de santé du joueur tombe à zéro. À chaque niveau complété, des points seront alloués au joueur. Chaque monstre abattu ainsi que chaque trésor obtenu augmentera aussi le pointage du joueur. The player may find swords that add to the treasure counter and increases performance in combat to reduce the amount of lost health in each encounter.

Environnement

Le joueur traversera une variété d'environnement, par exemple, une forêt où se situe l'entrée du donjon.

Structure

Avatar

L'avatar est animé pour quatre directions, chacun ayant trois positions, pour un total de douze images.

Environnement

Niveau

Les niveaux sont faits avec Tiled editor v. 0.8.0 (<http://www.mapeditor.org/>)

Layers

Les layers sont décrits dans l'ordre qu'ils doivent être dessiner à l'écran.

Floor

Type: tiles - Peut être traversé

Contient toutes les tuiles dont le sol est composé.

Walls

Type: tiles - Ne peut être traversé

Contient des tuiles qui bloquent le chemin des monstres et du joueur.

furniture

Type: tiles - Ne peut être traversé

Décorations.

Decoration

Type: tiles - Can be traversed if the layers below can also be traversed.

Décorations.

pickable

Type: tiles - Peut être traversé

Peuvent être ramassés par le joueur (ou les monstres) en marchant dessus.

interacted

Type: tiles - Ne peut être traversé - State 2 layer

Layer spécial qui fonctionne avec le layer *Interactable*. Seulement une des layer est affiché a la fois dans une tuile. Le joueur interagit avec la layer *Interactable* et cause celle-ci a apparaitre.

interactable

Type: tiles - Ne peut être traversé

Tout objets destinés a être activés par le joueur.

Groups

monsters

Type: objects - Ne peut être traversé

Contient le point de départ pour tout les monstres.

triggers

Type: objects - Peut être traversé

Contient la position de tout objet capable d'activé un événement comme un changement de niveau.

Interface Utilisateur

Le menu principal est sous forme d'un niveau.

Instructions et Objectifs

Survivre l'expédition dans le donjon et amasser des richesses!

- 1- Déplacez votre avatar avec la manette.
- 2- Attaquez les monstres avec le bouton.
- 3- Intéragissez avec les objets en utilisant le bouton de la manette également.

HUD

Affiche le compteur de points de vie et de points.

Audio

Les effets sonores sont utilisé lorsque le joueur est atteint par un monstre ainsi que le contraire et lorsqu'un trésore est obtenu. Une grande variété de sons son incorporés au jeu (la plupart ne sont pas utilisés).

Index

<http://compsci.ca/v3/viewtopic.php?t=10908>)

<http://compsci.ca/holtsoft/OOTRef.pdf>)

Language reference

<http://compsci.ca/holtsoft/OOTRef.pdf>

Code Source

Location of the source code is on Assembla.com

<https://subversion.assembla.com/svn/crossingpoint/>

Open Turing peut être trouvé ici:

<http://tristan.hume.ca/openturing/>

ou un autre éditeur Turing ici:

<https://github.com/downloads/Open-Turing-Project/turing-editor-qt/OpenTuring.zip>

Forums Open Turning: <http://compsci.ca/v3/index.php?h=1&pf=120>

exemple pour l'animation

<http://compsci.ca/v3/viewtopic.php?t=18100>

functions and processes:

<http://compsci.ca/v3/viewtopic.php?t=14665&highlight=function>

Classes tutorial

http://wiki.compsci.ca/index.php?title=Turing_Classes_Part_I&redirect=no

<http://compsci.ca/v3/viewtopic.php?t=10905>

<http://compsci.ca/v3/viewtopic.php?t=10908>

Resources

<http://vxresource.wordpress.com/>

<http://www.alteredgamer.com/rpg-maker/105824-some-new-tilesets-vx/#>

http://wiki.compsci.ca/index.php?title=Turing_File_IO

<http://forums.rpgmakerweb.com/index.php?/topic/5938-grannys-lists-faerie-magic-and-mushrooms/>

<http://charas-project.net/resources.php?wa=0&lang=br&area=2&offset=127&howmany=10&fsearch=>

<http://ayene-chan.deviantart.com/art/RPG-Maker-Wooden-Floor-316114650>

Collections:

<http://compsci.ca/v3/viewtopic.php?t=20302&highlight=collections>

Extrait de Code Source

Loading multiple Pic

Source: <http://compsci.ca/v3/viewtopic.php?t=21335>

```
%http://compsci.ca/v3/viewtopic.php?t=4637&highlight=gif
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Program To Use GIFs %
% Kevin The Short One %
% Canada Flag Animated%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
View.Set ("offscreenonly")
colorback (black)
cls    %% make the screen look good...
%%%%% ^ sets the window size etc, i used window.open so i can close at the end
var picnum, pic : int := 0
%%% u need a variable for the picture.. and one to count up the frames
loop
    exit when hasch  %% exits when u push a button
    if picnum not= 8 then  %% the number of frames, makes sure it done exceed
        picnum += 1  %% makes the frame count go higher
    else  %% the frame number is higher then the amount of pictures
        picnum := 1  %% restart at frame 1
    end if
    pic := Pic.FileNew ("Images/can" + intstr (picnum) + ".BMP")
    %% ok... that makes the picture variable change to reflect the current picture
    %% intstr converts interger to string so it will be can1.bmp or can2.bmp depending
    on the frame...
    Pic.Draw (pic, maxx div 2 - 75, maxy div 2 - 50, picCopy)  %% draws the flag in the
    middle of screen
    View.Update
    cls
    delay (100)  %% so its kinda smoother... u could use View.Update **i made this for
    simplicity..
end loop
```

Images

Turing requires colorkeyed images (similar to 1bit alpha). Since doing colorkey inside GIF or JPEG caused us some problems, we went for 24bits BMPs.

The color key selected is pure green; 0x00ff00 or (0,255,0)

Image Slicing

Slicing was done in GIMP 2.8. Character sheets are used and are composed of 2 rows of 4 character sheets each.

Each character sheets are 3 columns and 4 rows. Each rows indicate one direction(N,S,E,W) and each columns the left footstep, idle and right footstep.

Image conversion

Converting .png into .bmp

Using ImageMagick, we can batch convert png inside a folder to bmps.

```
C:\ImageMagick\magick.exe -format bmp *.png
```