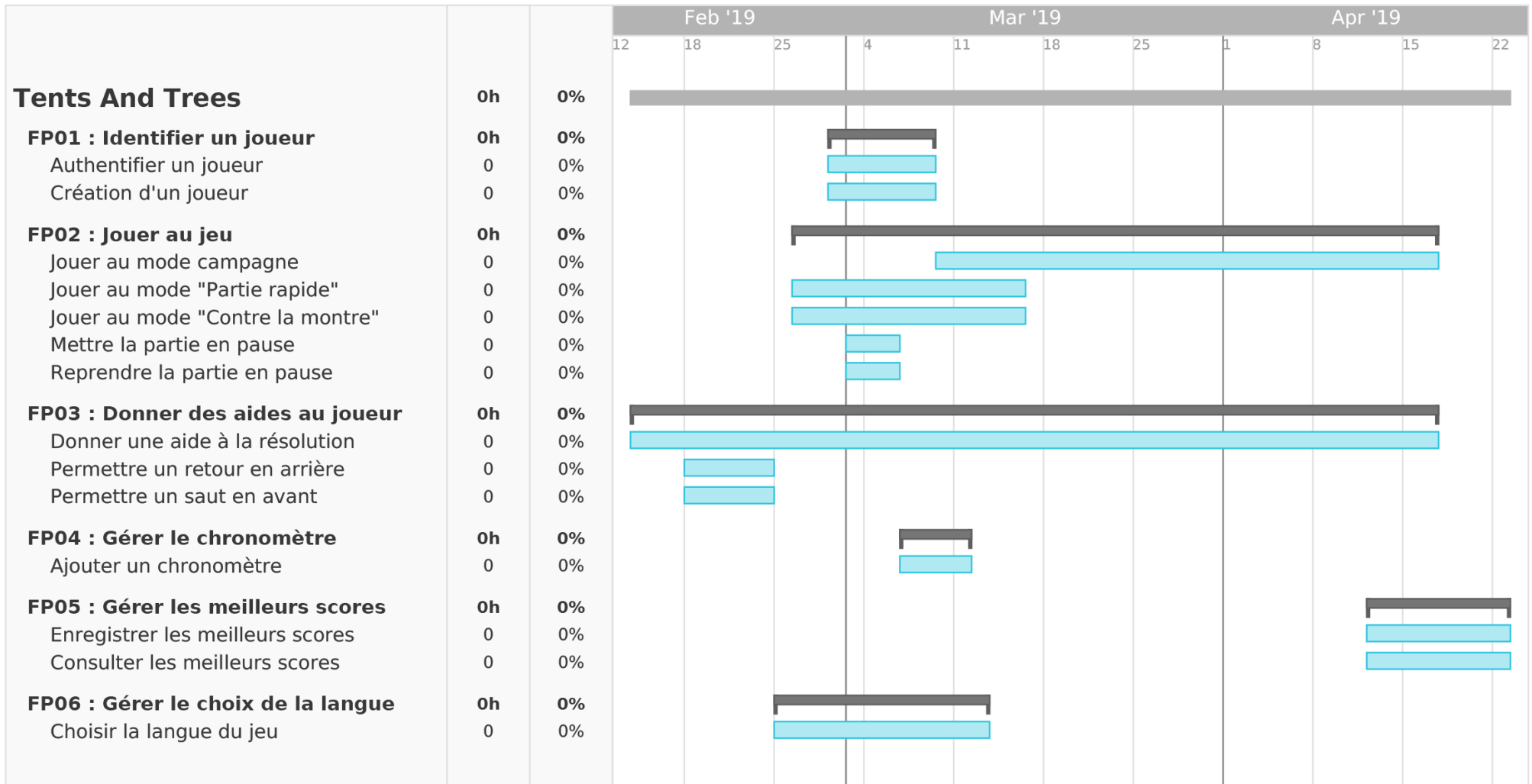


TENTS & TREES

BOUYEURE Lucien & TOUZE Maxime & PETIT Corentin & HUAULME Corentin &
TOURPE Florian & GONCALVES Quentin & VAIDIE Camille

Organisation du travail



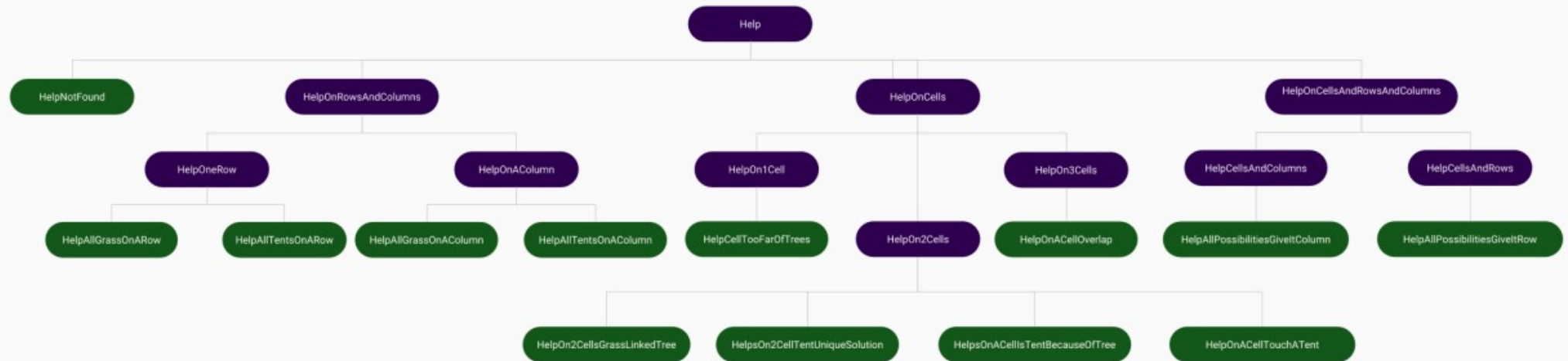
Choix de Conception

- Modèle vue contrôleur
- Game observée par l'écran de jeu
- Classes helper : inspiration du système de compilation
- Classes d'aides : chacune a son comportement, implémenté façon exceptions Java

Hiérarchie des aides

class Help

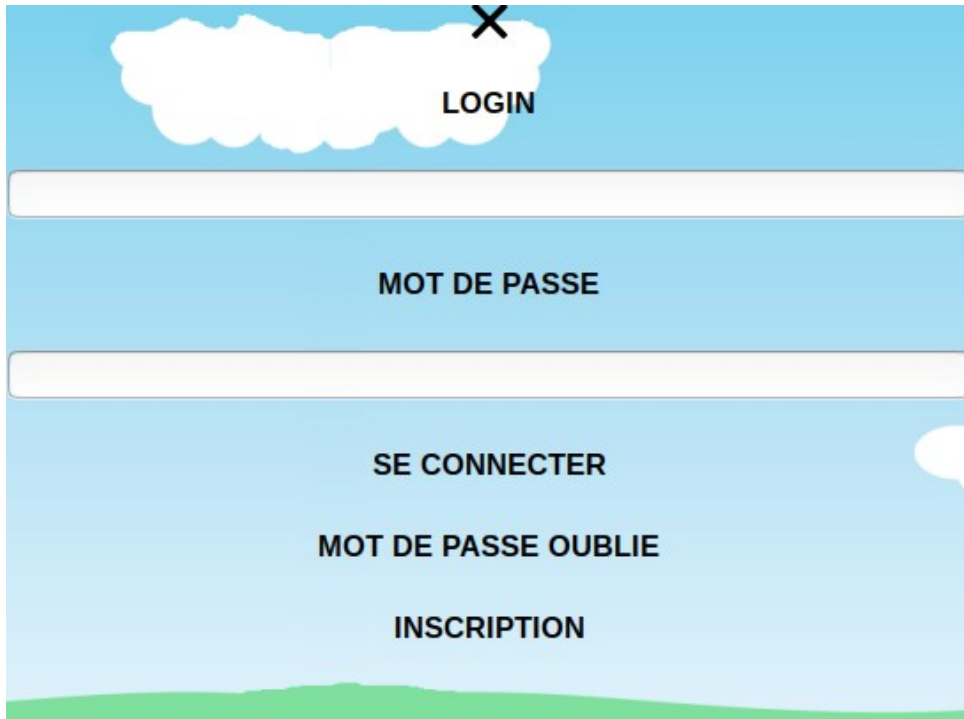
The help class is an abstract class. It is the general parent of all the other help classes in the game. To see the help classes tree, see below :



Fonctionnalités importantes



Connexion et mot de passe



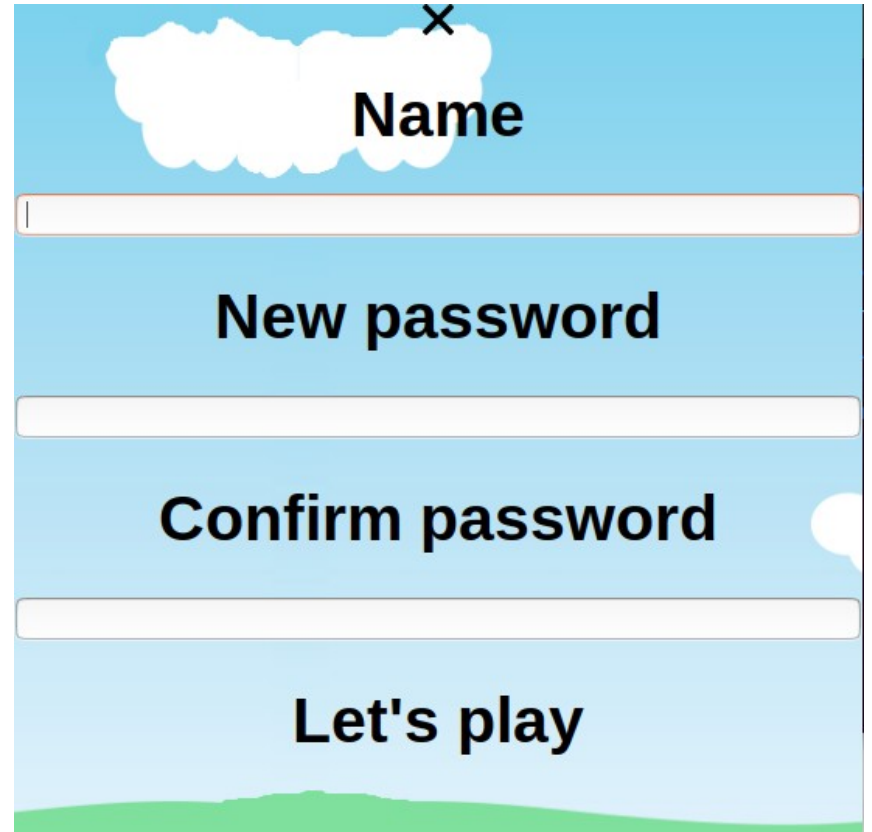
LOGIN

MOT DE PASSE

SE CONNECTER

MOT DE PASSE OUBLIE

INSCRIPTION



Name

New password

Confirm password

Let's play

Table des meilleurs scores



Aides nombreuses et précises



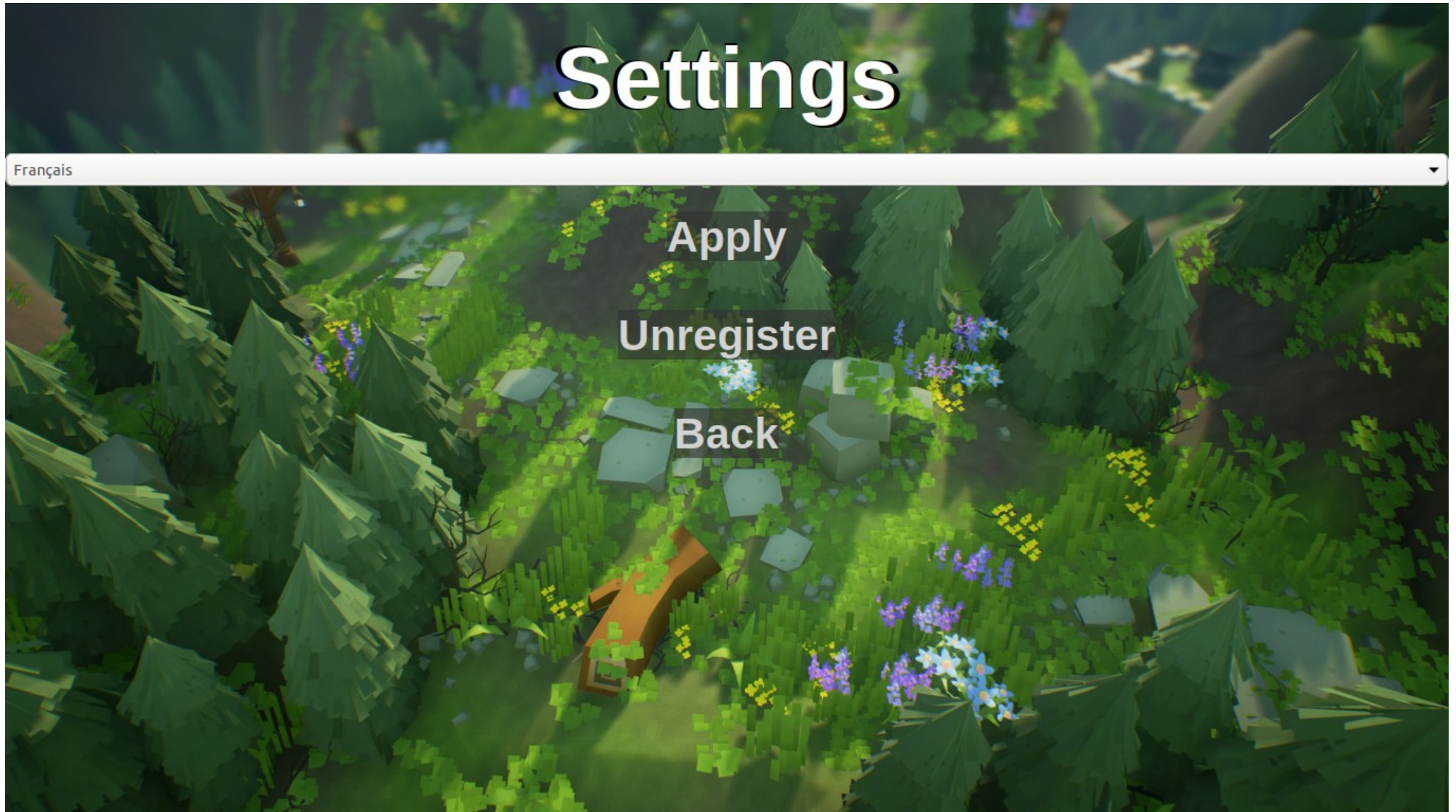
Aides sur plusieurs niveaux



Plusieurs modes de jeu



Pourquoi acheter notre jeu ?



Differentes images (arbre, tente)

3 0 2 0 2 0

0

3

0

2

1

1

00h 00m 17s

Undo Redo

Reset Grid

New Hypothesis

Refute Hypothesis

Help

Mode aventure et étoiles



Hypothèse de jeu



RDoc

Home

Pages Classes Methods

Search

Parent

Object

Methods

```
new
# ==
# cellList
# getRes
# getHelpText
# helpFound?
# price
# to_s
```

class Help

The help class is an abstract class. It is the general parent of all the other help classes in the game. To see the help classes tree, see below :

The classes in blue are abstract while the ones in green are concrete.

This class' direct children are the `HelpOnRowFound` class and the abstract classes `HelpOnRowsAndColumns`, `HelpOnCells` and `HelpOnCellsAndRowsAndColumns`. This class knows the methods below :

Public Class Methods

new()

The class' constructor. This method takes no parameters and initializes two variables :

- `price` : An integer giving the cost of each help. Here, it is initialized at 5.
- `helpText` : The text that will be shown every time the player needs help.

Public Instance Methods

==(otherHelp)

This method is a redefinition of the `==` method.

Returns

```
return (self.class == otherHelp.class)
```

A boolean telling if the two classes are equal.

cellList(helpLevel)

This method returns an array with all the important cells for the help.

getRes(helpLevel)

This method returns the UI printable form of the help.

getHelpText(helpLevel)

This method fetches the help text for a given help level. It is used to initialize the `@helpText` variable in the class' constructor.

Returns

Returns the help text.

helpFound?()

This tries to find the necessary help.

Returns

Returns true if help has been found, else it returns false.

price(helpLevel)

This method checks the price of a given help level.

Returns

```
price = @priceLevel
```

The `@price` variable (initialized at 5) times the help level, so the prices go higher as the help gets higher and the game gets easier.

to_s()

This method a redefinition of the `to_s` method, used to print the help text.

Returns

Returns the `@helpText` variable's value and a line break.

Version:
Generated by RDoc 4.1.1.
Based on Darfleh by Michael Granger.

15

Tutoriel

The screenshot displays a game interface with a 6x6 grid on the left and a menu overlay on the right. The grid contains tree icons on green squares, with numbers 1, 2, and 3 indicating their levels. The menu overlay includes buttons for 'Pause', 'Undo Redo', 'Reset Grid', 'New Hypothesis', 'Refute Hypothesis', 'Help', and 'Next'. A red text message at the bottom right reads: 'Welcome to the tutorial level, to continue, click on "Next".'

	2	1	2	1	1	2
3	Tree (Level 3)		Tree (Level 2)		Tree (Level 1)	
0	Tree (Level 0)					
3		Tree (Level 1)				Tree (Level 1)
0						Tree (Level 1)
2						
1	Tree (Level 0)			Tree (Level 2)		

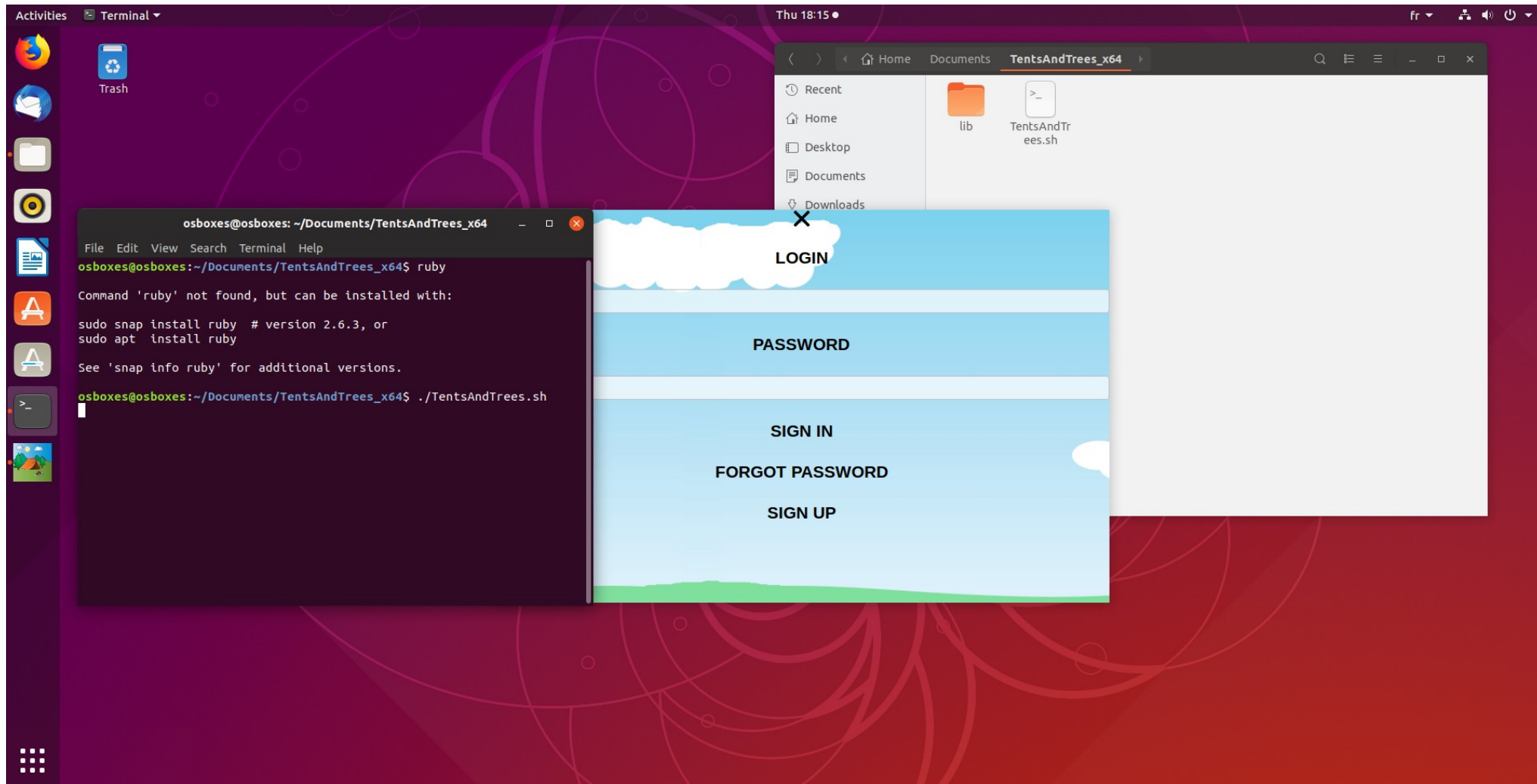
Pause
Undo Redo
Reset Grid
New Hypothesis
Refute Hypothesis
Help
Next

Welcome to the tutorial level, to continue, click on "Next".

Conforts de jeu



Exécution



BON JEU