UML Code

```
class Filter {

}

class Main {
  FileReader fileReader;
  1 -> 1 Maze maze;
  1 -> 1 MainFrame mainFrame;
  1 -> 1 Game game;
  TimeLimit timeLimit;

}

class Position {
  int x;
  int y;
}

class TimeLimit {
  Timer timer;
  int time;
}

class DisplayPanel {
```

```
 int chipsLeft;
 boolean gamePaused;
 Game game;
}

class InterfacePanel {
 JLabel levelLabel;
 JLabel timeLabel;
 JLabel chipsLeftLabel;
 JTextField levelField;
 JTextField timeField;
}


class InventoryPanel {
 Game game;
}

class MainFrame {
 1->1 DisplayPanel displayPanel;
 1->1 InterfacePanel interfacePanel;
 1->1 InventoryPanel inventoryPanel;
 1->1 RulesPanel rulesPanel;
 1->1 TextPanel textPanel;
 Filter filter;
 Game game;
 boolean paused;
```

```
  boolean refreshing;
  boolean replay;

}

class RulesPanel {
  JLabel text;

}

class TextPanel {
    JLabel text;
}



class Game {
  1->1 Maze maze;
  1->1 Chap;
  InfoField info;
  boolean onField;
  1->1 TimeLimit timeLimit;
  boolean finished;

}

class Maze {
    int currentLevel;
```

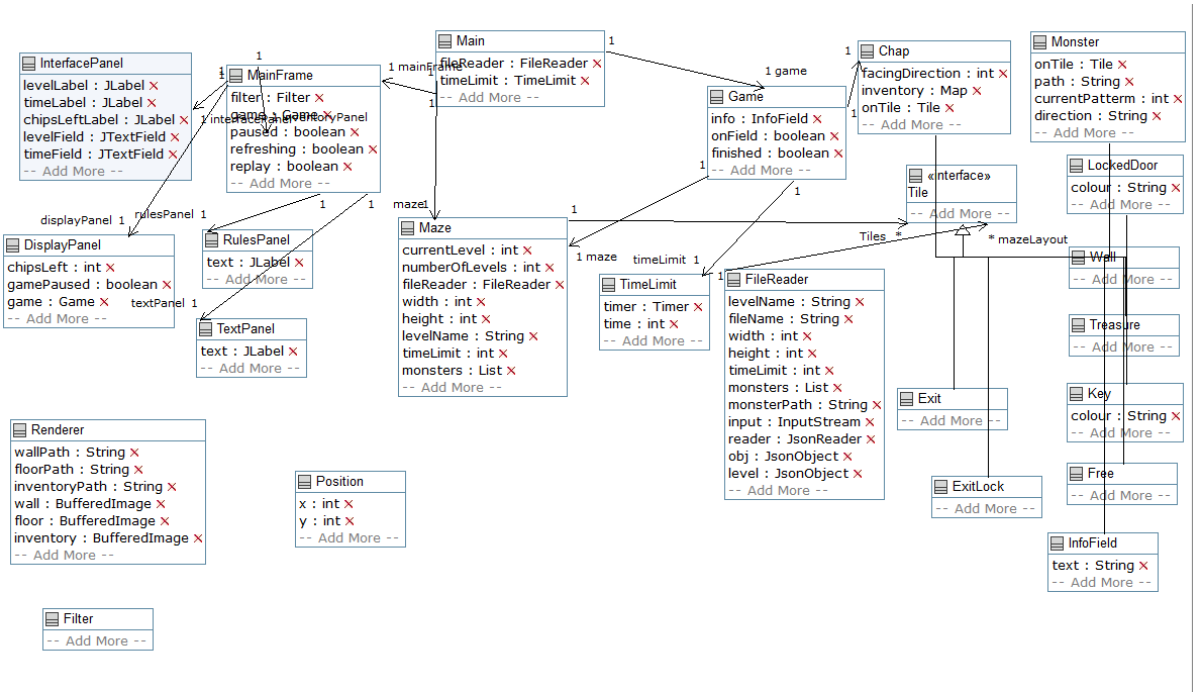
```
    int numberOfLevels;
    FileReader fileReader;
    1->* Tile Tiles;
    int width;
    int height;
    String levelName;
    int timeLimit;
    List<Monster> monsters;

}

interface Tile {
  boolean isSolid;
  Position currentPosition;

}

class Chap {
  isA Tile;
  int facingDirection;
  Map<String,String> inventory;
  Tile onTile;

}

class Exit {
  isA Tile;
```

```
}

class ExitLock {
  isA Tile;

}

class Free {
  isA Tile;
}

class InfoField {
    isA Tile;
    String text;

}

class Key {
    isA Tile;
    String colour;
}

class LockedDoor {
  isA Tile;
  String colour;
}
```

```
class Monster {
  isA Tile;
  Tile onTile;
  String path;
  int currentPatterm;
  String direction;
}

class Treasure {
  isA Tile;

}

class Wall {
  isA Tile;

}

class FileReader {
  String levelName;
  String fileName;
  int width;
  int height;
  int timeLimit;
  1->* Tile mazeLayout;
  List<Monster> monsters;
```

```
  String monsterPath;
  InputStream input;
  JsonReader reader;
  JsonObject obj;
  JsonObject level;



}

class Renderer {
  String wallPath;
  String floorPath;
  String inventoryPath;

  BufferedImage wall;
  BufferedImage floor;
  BufferedImage inventory;

}
```

Diagram


InterfacePanel
levelLabel : JLabel
timeLabel : JLabel
chipsLeftLabel : JLabel
levelField : JTextField
timeField : JTextField
-- Add More --
MainFrame
filter : Filter
game : Game
paused : boolean
refreshing : boolean
replay : boolean
-- Add More --
Main
fileReader : FileReader
timeLimit : TimeLimit
-- Add More --
Game
info : InfoField
onField : boolean
finished : boolean
-- Add More --
Chap
facingDirection : int
inventory : Map
onTile : Tile
-- Add More --
Monster
onTile : Tile
path : String
currentPatterm : int
direction : String
-- Add More --
«interface»
Tile
-- Add More --
LockedDoor
colour : String
-- Add More --
DisplayPanel
chipsLeft : int
gamePaused : boolean
game : Game
-- Add More --
RulesPanel
text : JLabel
-- Add More --
TextPanel
text : JLabel
-- Add More --
Maze
currentLevel : int
numberOfLevels : int
fileReader : FileReader
width : int
height : int
levelName : String
timeLimit : int
monsters : List
-- Add More --
TimeLimit
timer : Timer
time : int
-- Add More --
FileReader
levelName : String
fileName : String
width : int
height : int
timeLimit : int
monsters : List
monsterPath : String
input : InputStream
reader : JsonReader
obj : JsonObject
level : JsonObject
-- Add More --
Wall
-- Add More --
Treasure
-- Add More --
Key
colour : String
-- Add More --
Free
-- Add More --
Exit
-- Add More --
ExitLock
-- Add More --
InfoField
text : String
-- Add More --
Renderer
wallPath : String
floorPath : String
inventoryPath : String
wall : BufferedImage
floor : BufferedImage
inventory : BufferedImage
-- Add More --
Position
x : int
y : int
-- Add More --
Filter
-- Add More --
displayPanel
rulesPanel
textPanel
maze
mainFrame
game
timeLimit
Tiles
mazeLayout