# A Tactical RPG Engine

Bilal Hussain

October 6, 2011

## 1 Introduction

A RPG (Role Playing Game) is game where player assumes the role of on the charac-
ter. A RPG is usually story driven and the character usually has quest to complete.
In the course of the game the player will go to different environments such as town
and dungeons. In they these environments the player will have to battle opponents
in battles. Combat in RPGs is normally a simple turn based system where player
and the opponents take turns attacks each other using various skills.

A Tactical RPG is sub-genre of an RPG that focuses on the combat side of the
genre. A Tactical RPG is series of `battles`, which take place in various environments
intertwined with an over-aching story.

Each `battle` is grid based (like chess) where each player has a number of units(pieces).
The players taken turns to moves their units. Each unit has attributes associated
with it such as strength, and hit points that affect all the actions in the game. Like
chess there are different kinds of units which affects how the unit moves and what
action they can perform. A unit can attack other player's units, the goal of the battle
is usually to defeat all the opponents units.

The aim of this project is to create a engine which will take resources such as
graphics, sound and rules of the game to create a runnable Tactical RPG.

## 2 Objectives

### 2.1 Primary

- To develop an engine that takes:

    - The definition of character attributes and a combat system.
    - The definition of a world broken up into the smaller environments.
    - The definition of simple story as a wrapper for the whole game, from
      the start to the conclusion of the game

and create a playable tactic RPG.

- To include in the engine support for the following:

  - `units` with associated attributes such as:
    * Hit-points (which represent the health of the unit).
    * Strength.
    * Defence.
    * Move (The number of tiles the unit can move each turn).
  - `battles` which take place on grid and include:
    * A set number of `units` for each player.
    * `combat` between adjacent units.
      · When the unit hit-points are reduced to zero they are `defeated` and are removed from the map.
    * A Winning `condition` such as defeat all of the other players units.
    * Battles are `turn based` meaning that each player moves all their units (once) before the next player turn.
  - Movement between different environments.
  - A set of behaviours for how the non-player characters should behave.
    * Including pathfinding.
  - A graphical representation of of the game.
    * Which is show the grid with all the units.
    * Allow the user to move their units and see the opponents moves.
    * Allows the user to attack the opponents units.

## 2.2   Secondary

- Tile `height`, where units can only move to tiles of a smilier height.

- Tiles that are not passable such as sea, lava, etc.

- Tiles have different movement costs associated with them.

- Isometric graphics view of the game.

- Long distance weapons/magic for player and AI.

- Direction and height of the character's tile affects attack.

- Sound effects.

- Music.

- Saving and loading games.

- A default set of behaviours for non-player characters.

## 2.3 Tertiary

- Custom events

  - Attached to units or titles, could be used for:
    * Making the player win if some enemies unit has less then 50% Hit Points.
    * Damaging a character if step on a specified.
    * Showing some part of the story when a player's character reach a specified tile.

- A graphical editor for making custom maps and events.

- Healing item/skills.

- Animations for units, combat and movement.

## 2.4 Ethical Considerations

- Form by 28th October.

- Collection of data from questionnaire.

  - Just result of questionnaire, no personal data.

- Asking users to create a game.

- Asking users to play the created game.

## 2.5 Resources

- None.