Group#9

# DeadPool Rush; A game for legends

Requirements:

The aim is to make a create a simple 2.5D running game in which a super hero character is running to find the villain and save his girlfriend from him. The character will face obstacles and enemies to on his way, along with guns and power ups on the ground to pick up and kill the enemies.

Features:

- Weapons:
  - Multiple types of guns
  - Jetpack
  - Swords
  - Bombs
- Enemies:
  - Different enemies on different scores
  - Boss enemies on the end of every stage
  - Enemies with Guns
- Stages:
  - At least 5 different stages with different enemies, guns, obstacles and power ups

Storyline:

When it comes to super heros, DeadPool is the one who'll beat'em all. DeadPool is going to save his girlfriend Vennesa from Francis!

While rushing to kill Francis, DeadPool accidentally threw his duffel bag. Help him find his guns and power ups while going through the army of Francis!

- Answer the following questions about your project:
  - Q#1: Do you think you need mathematical verification of correctness of your system or a part of your system? Why?

Answer: No, as we’re making a game, so the inputs of our system will not exceed.

- Q#2: Can you separate various concerns of your project from functional and quality perspectives? Highlight the concerns and describe how can you handle concerns separately.
  Answer: There’s a team of 4 people behind this project, so division of work would be done according to everyone’s strength which will help us separate the parts of this system. For example, we’ve separated the concerns in the following way:
- Member #1: Graphics and Animations of Character and Background
- Member #2: Backend Development of Character, Pick-up objects & Obstacles
- Member #3: Backend Development of Enemies and Stages/Environment
- Member #4: Integration of inputs of the other 3 members into a single system

- Identify some functional modules in your system. Discuss coupling and cohesion aspects.
  Our game would have following modules:
  - Game Character
  - Background
  - Pickup Objects
  - Enemies
  - Obstacles

  From the perspective of Coupling, the part of the modules that will communicate with the other modules would be the interactions; the functions being triggered when objects seem to be touching each other on screen. Whereas, the cohesion among the functions and objects would be in a way that the functions that must be triggered when the fire button is pressed say, must aslo.

- Identify the potential future changes in your system. Pick one potential change and discuss how would you address it in your system?
  The potential future changes in our game would be totally dependent upon current situation in the real world. In gaming, it’s the best approach to provide the users an environment that is close to the real one, which also includes the environmental changes, events and other things happening around the world. For example, our character can suddenly enter into a zone hit by Irma storm or it may go to the jungles of Amazon to find its lost thing and faces the dense jungle to go through, or it may be new year where there’s firecrackers and explosives everywhere to help him blow his enemies etc.

- Which increments would you suggest if you are asked to build your system incrementally?

  In Incremental approach, we will make the software by building one feature of the software from the requirements at a time. Thus, keep on adding the pieces until it’s complete. First one module of the software will be developed and tested. Then Second module of the software will be developed and tested and then it will be integrated with first module then third module when me developed and tested and will be integrated with the first two module and this way we’ll keep adding modules until software is complete.