**T****rabalho de Arquitetura de Software**

*Gabriel Sousa Kraszczuk* **– *Prontuário 135025-1***

*Francisco Gonçalves da Mota Longhini* **– *Prontuário 135025-1***

*O Diagrama de Classe Original, antes das alterações, se encontra na pasta* ***Docs*** *do projeto Java, com o seguinte nome “****DiagramaDeClasse – Versão Original****”*

*O Diagrama de Classe Modificado, depois das alterações e implementações realizadas por este trabalho, se encontra na pasta* ***Docs*** *do projeto Java, com o seguinte nome “****DiagramaDeClasse – Versão Nova****”.*

**IMPLEMENTAÇÕES:**

Para o nosso trabalho, foi atribuído as seguintes implementações:

*-* ***Múltiplas Vidas*** *– Agora o personagem não morre ao simplesmente entrar em contato com a água, ele apenas perde uma vida.*

*-* ***HUD –*** *Interface gráfica que permite ao jogador visualizar a quantidade de pontos coletados e a quantidade de vida restante.*

Entretanto, outras funcionalidades também foram implementadas no projeto, são elas:

*-* ***Sprites dos Personagens –*** *Permite a troca da imagem do personagem, dependendo da direção que o mesmo está se movimentando.*

*-* ***Novos Elementos –*** *Foram adicionados novos elementos no mapa, tais como: Terra, Lava, Terra de Caverna, Placa, Pedra, Chão de Pedra e Parede Decorativa, onde cada elemento possui uma propriedade de ser transponível ou não.*

*-* ***Cavernas*** *–* *Foi implementado uma estrutura de mapas, que permite a adição de cavernas ao jogo.*

*-* ***Múltiplos Mapas e Transição*** *– Foi implementado uma estrutura que permite a transição de mapas, ou seja, quando se está em um mapa e se anda até o estremo do mapa (cima, baixo, esquerda e direita), caso haja um mapa nesta direção, é realizada a transição de mapas.*

Para a implementação dos componentes acima, foram necessárias efetuar a criação das seguintes classes:

* ***MULTIPLOS MAPAS E TRANSIÇÃO*** *– Foi necessário criar a classe Mapas e a classe EstruturaMapas, que são as responsáveis por gerenciar o mapa, e permitir a interligação de vários mapas, de acordo com a sua posição, ou seja, por exemplo: quando o personagem caminha até o extremo do mapa atual, caso haja um próximo mapa, o mesmo será carregado.*
* ***SPRITES DO PERSONAGEM*** – *Foi necessário criar uma classe Movimentacao, que é a responsável por gerenciar o movimento do personagem no jogo, com base nos elementos adicionados na classe enum Elemento.*
* ***PONTOS DE VIDA –*** *Para adicionar o hud e os pontos de vida, foi criada uma classe chamada Hud, responsável por gerenciar a vida e os pontos, como também de gerar a tela para ser exibida, posteriormente pela classe TelaJogo.*
* ***DEMAIS ALTERAÇÕES*** *– Para que fosse possível a interligação de tantas implementações, as classes Tabuleiro, TelaJogo, Main, Elemento e SaidaJogo, precisaram sofrer algumas alterações, muitas delas, com a adição de campos, para a interligação dos objetos das classes citadas acima, como também na criação de métodos que permitem a atualização da tela para o jogador. Também houve alterações na estrutura da classe tabuleiro, onde o método* ***fazerMovimento()*** *foi transformado em uma classe* ***Movimento.***