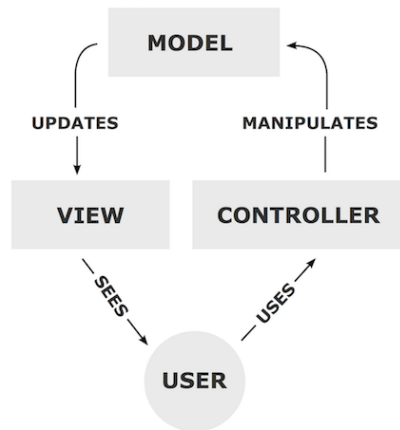


Design of behavioural aspects:



O jogo apresenta 3 packages principais:

- **Model:** Expressa o comportamento da aplicação. Gerência diretamente os dados, a lógica e as regras da aplicação.
- **View:** Interface do jogo e representação de toda a informação do Model como output.
- **Controller:** Aceita os inputs e converte-os para o Model e para o View.

Design Patterns:

O nosso projeto usará Singleton (classe `GamePlay`) para assegurar que a classe do jogo apenas terá uma instância e poderá ser acessível por todos. Usará também o pattern STATE (classe `GameLevel`) visto que no jogo constam 3 níveis em que o herói tem que vencer.

Também será utilizado o Template Method (classe `GamePlay`) para facilitar o desenvolvimento do jogo.

GUI Design

O nosso jogo começará por apresentar um menu principal, com 4 opções:

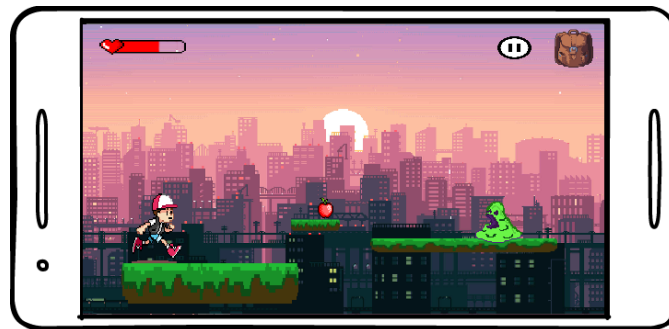
- Play;
- Settings;
- Score;
- Exit;



Pressionando o botão Play somos redirecionados para o jogo.

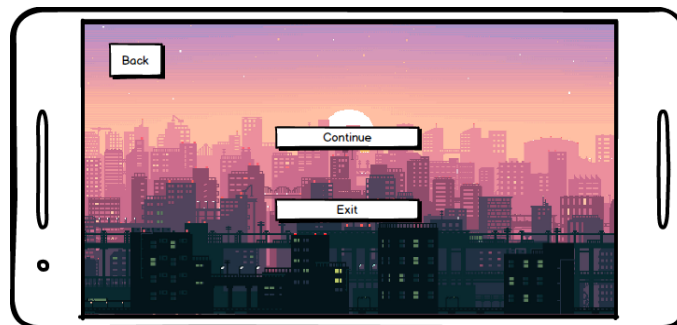
No canto superior esquerdo encontra-se a vida do herói que vai decrementando à medida que for sofrendo dano. No canto superior direito, encontramos o item de pause e da “bag”.

Ao clicarmos sobre a mochila os itens recolhidos pelo herói ao longo do nível vão aparecer, e poderão ser selecionados. (exemplo: maçã -> regenera o herói com alguns pontos de vida; Pistola-> o herói fica armado e poderá destruir os inimigos) O herói move-se através da utilização do acelerómetro do telemóvel.



Ao clicarmos sobre o botão de pause somos redirecionados para o menu de pause. Aqui podemos pressionar:

- Botão Back-> voltar ao menu principal
- Botão Continue->continuar a partida
- Botão Exit ->sair do jogo;



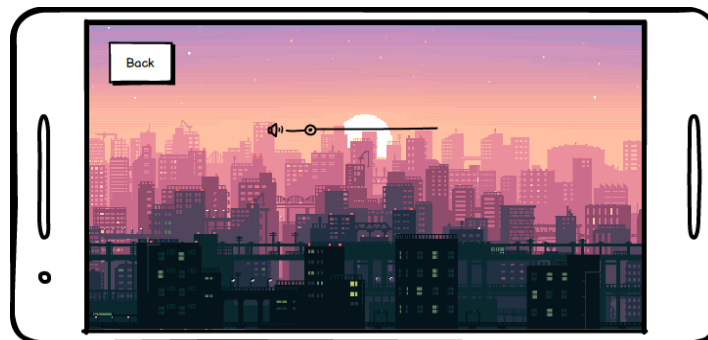
Se a vida do herói chegar ao fim somos levados para o menu de Game Over. Onde apenas podemos voltar para atrás, ou seja para o menu principal.



Se o jogo for concluído com sucesso, ou seja acabar o nível ainda com vida, o jogador será redirecionado para o menu de Vitória



Se no menu principal selecionarmos a opção Settings, somos redirecionados para o menu das opções. Onde podemos alterar o volume da música do jogo. Ao pressionarmos o botão Back voltamos para o menu principal.



Se no menu selecionarmos a opção Score, somos redirecionados para o menu das pontuações onde serão listadas as pontuações do jogador com a respectiva data. Ao pressionarmos o botão Back voltamos para o menu principal



Test Design

Alguns dos testes unitários que iremos utilizar para testar e guiar o nosso código:

Testes aos movimentos:

- `JumpTest();`
- `MoveLeftTest();`
- `MoveRightTest();`
- `BulletMovementTest();`
- `DamageMovementTest();`

Testes aos itens encontrados pelo caminho de jogo:

- `AppleTest();`
- `WaterTest();`
- `Etc..`

Testes às colisões:

- `ColisionPlatformTest();` (testes para os diferentes tipos de colisões com as diferentes plataformas)
- `ColisionHeroAlienTest();`

Testes aos estados (WIN, LOST, PLAYING, etc..)

- `LoseGamaByFallTest();`
- `LoseGameTest();`
- `VictoryGameTest();`
- `NextLevelTest();`
- `ScoreUpdateTest();`
- `PauseGameTest();`

Teste á física do jogo

- `GravityTest();`
- `VelocityTest();`
- `FrictionTest();`