**Cronograma de Montagem - Caça ao Monstro 2**

**Dia 1: Clonagem e Preparação Inicial (4 horas)**

1. **Clonagem do Projeto Original (0h30)**
   * Clonar o repositório do **"Caça ao Monstro"**.
   * Configurar a nova estrutura do projeto com o nome **"Caça ao Monstro 2"**.
   * Atualizar o README.md inicial com a descrição da nova versão do jogo.
2. **Planejamento de Mídias (Imagens e Sons) (1h30)**
   * Definir quais imagens e sons serão necessários para o jogo (jogador, monstro, grade, sons de fundo e efeitos).
   * Planejar o uso das bibliotecas gráficas e sonoras (por exemplo, **Pygame**).
3. **Configuração Inicial do Projeto (1h)**
   * Configurar o ambiente virtual e dependências (como a instalação de **Pygame** ou outras bibliotecas necessárias).
   * Adicionar novos arquivos de configuração como requirements.txt e setup.py atualizados.
4. **Estrutura de Arquivos e Diretórios (1h)**
   * Criar as pastas assets/ com subpastas para imagens e sons.
   * Preparar arquivos para novos módulos como assets\_manager.py, sounds\_manager.py, e display\_manager.py.

**Dia 2: Implementação das Imagens (6 horas)**

1. **Desenvolvimento do Módulo assets\_manager.py (2h)**
   * Implementar a lógica para carregar as imagens do jogador, monstro e fundo da grade.
   * Criar métodos para gerenciar e atualizar as imagens durante o jogo.
2. **Integração com o game.py e grid.py (2h)**
   * Modificar o fluxo do jogo para incluir a exibição das imagens usando o novo assets\_manager.py.
   * Renderizar a imagem de fundo da grade e atualizar as posições do jogador e monstro com suas respectivas imagens.
3. **Testes Visuais Iniciais (1h30)**
   * Testar se as imagens estão sendo carregadas e exibidas corretamente nas posições apropriadas.
   * Ajustar a lógica de renderização e movimentação para garantir que o jogo funcione visualmente conforme esperado.
4. **Ajustes Finais no Layout Visual (0h30)**
   * Ajustar o layout da grade e o posicionamento das imagens para garantir uma boa visualização do jogo.

**Dia 3: Implementação dos Sons (5 horas)**

1. **Desenvolvimento do Módulo sounds\_manager.py (2h)**
   * Implementar a lógica para carregar e tocar os sons (efeitos e música de fundo).
   * Criar métodos para reproduzir sons específicos durante eventos do jogo (captura, fuga, movimento).
2. **Integração dos Sons com o game.py (2h)**
   * Modificar o jogo para tocar música de fundo durante o loop principal.
   * Adicionar efeitos sonoros para eventos específicos, como a captura do monstro ou a movimentação do jogador.
3. **Testes de Sons (1h)**
   * Testar se os sons estão sendo reproduzidos no momento correto (movimento, captura, fuga).
   * Verificar se há sincronização adequada entre os eventos visuais e os efeitos sonoros.

**Dia 4: Testes e Depuração (6 horas)**

1. **Testes Funcionais dos Novos Recursos (3h)**
   * Testar o jogo completo com as novas funcionalidades de imagens e sons.
   * Verificar se as imagens são carregadas corretamente em diferentes condições e cenários.
   * Testar a música de fundo, efeitos sonoros e sua interação com o fluxo do jogo (foco no balanceamento dos sons).
2. **Testes Unitários dos Novos Módulos (2h)**
   * Escrever testes unitários para assets\_manager.py, sounds\_manager.py e display\_manager.py (mockando funções se necessário).
   * Garantir que os novos recursos estão funcionando corretamente sem quebrar a lógica anterior do jogo.
3. **Correção de Bugs (1h)**
   * Depurar e corrigir eventuais bugs detectados durante os testes, como problemas de renderização ou falhas na reprodução de som.

**Dia 5: Ajustes Finais e Polimento (5 horas)**

1. **Ajustes Visuais e Feedback do Usuário (2h)**
   * Melhorar a interface visual para o jogador (ex: adicionar uma tela de título, menus básicos).
   * Adicionar mensagens de feedback ao jogador durante a captura e fuga do monstro (ex: textos ou efeitos gráficos).
2. **Balanceamento do Jogo (1h)**
   * Ajustar a dificuldade do jogo (tempo de resposta do monstro, velocidade de movimento, número de turnos até o monstro escapar).
   * Fazer ajustes finos no tempo de reprodução de sons e efeitos para garantir uma experiência agradável.
3. **Documentação e Atualização do README.md (1h)**
   * Atualizar o README.md com instruções detalhadas de como jogar, novos recursos adicionados, e como instalar as novas dependências (como Pygame).
   * Adicionar um guia sobre como personalizar ou modificar as imagens e sons.
4. **Teste Final e Preparação para Entrega (1h)**
   * Executar o jogo finalizado várias vezes para garantir que todos os aspectos (gráficos e sons) estão funcionando conforme esperado.
   * Garantir que o projeto esteja pronto para ser clonado e executado facilmente por outros desenvolvedores ou jogadores.

**Resumo do Cronograma:**

| **Dia** | **Tarefa Principal** | **Tempo Estimado** |
| --- | --- | --- |
| 1 | Clonagem e Preparação Inicial | 4 horas |
| 2 | Implementação das Imagens | 6 horas |
| 3 | Implementação dos Sons | 5 horas |
| 4 | Testes e Depuração | 6 horas |
| 5 | Ajustes Finais e Polimento | 5 horas |

**Total: 26 horas**

**Considerações Finais:**

1. **Aproveitamento do Código**: A estrutura de código e lógica original é reaproveitada ao máximo, focando principalmente na adição de novos módulos (para gerenciamento de imagens e sons) e integração com o loop principal do jogo.
2. **Flexibilidade para Expansão**: Com essa estrutura, o jogo está pronto para receber outras melhorias, como menus interativos, novos efeitos gráficos ou até multiplayer, caso você queira expandir ainda mais o projeto no futuro.