PROTÓTIPO DE MOUSE UTILIZANDO ACELERÔMETROS: VERSÃO 2

Luiz Roberto Leicht
luizleicht@gmail.com

Orientador: Miguel Alexandre Wisintainer

Roteiro

- Introdução
- Objetivos
- Fundamentação teórica
- Desenvolvimento
- Resultados e discussão
- Extensões
- Conclusão
- Demonstração

Introdução

- Grande número de deficientes físicos
- Auxiliar na inclusão digital
- Extensão do trabalho desenvolvido por Jennrich (2010)

Objetivos

- Aperfeiçoar o trabalho desenvolvido por Jennrich (2010)
  - Integração com hardware
  - Utilizar novo acelerômetro
  - Permitir comunicação via conexão USB
  - Ser multiplataforma
  - Permitir calibração de movimentos
  - Aprimorar cliques

Fundamentação teórica

- Conceitos básicos
  - Deficiência física
  - .NET Micro Framework 4.1
  - Dispositivo FEZ Domino
    - USB Host
    - SD
    - HID
  - Acelerômetro eZ430-Chronos
  - Protótipo de mouse utilizando acelerômetros Jennrich (2010)
- Trabalhos correlatos
  - Tongue Drive
  - Mouse Visual Bradesco
  - Capacete EPOC

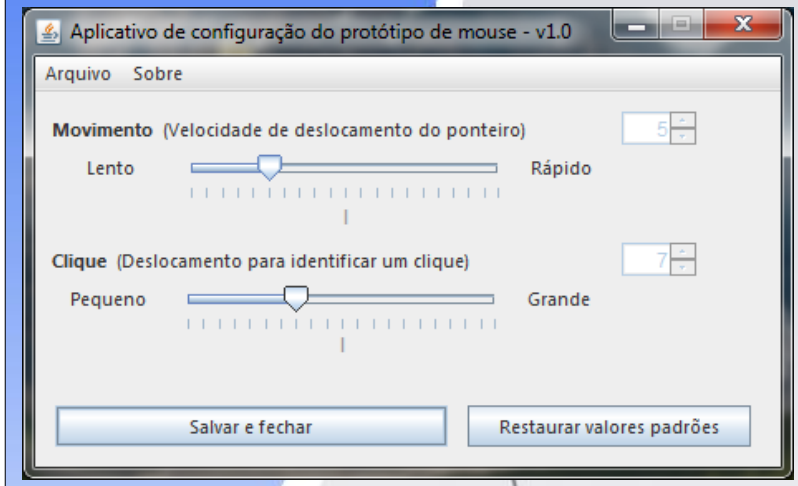
Desenvolvimento

- Requisitos funcionais
  - Interpretar movimentos do acelerômetro.
  - Executar comandos de um mouse tradicional.
  - Possuir interface de configuração.

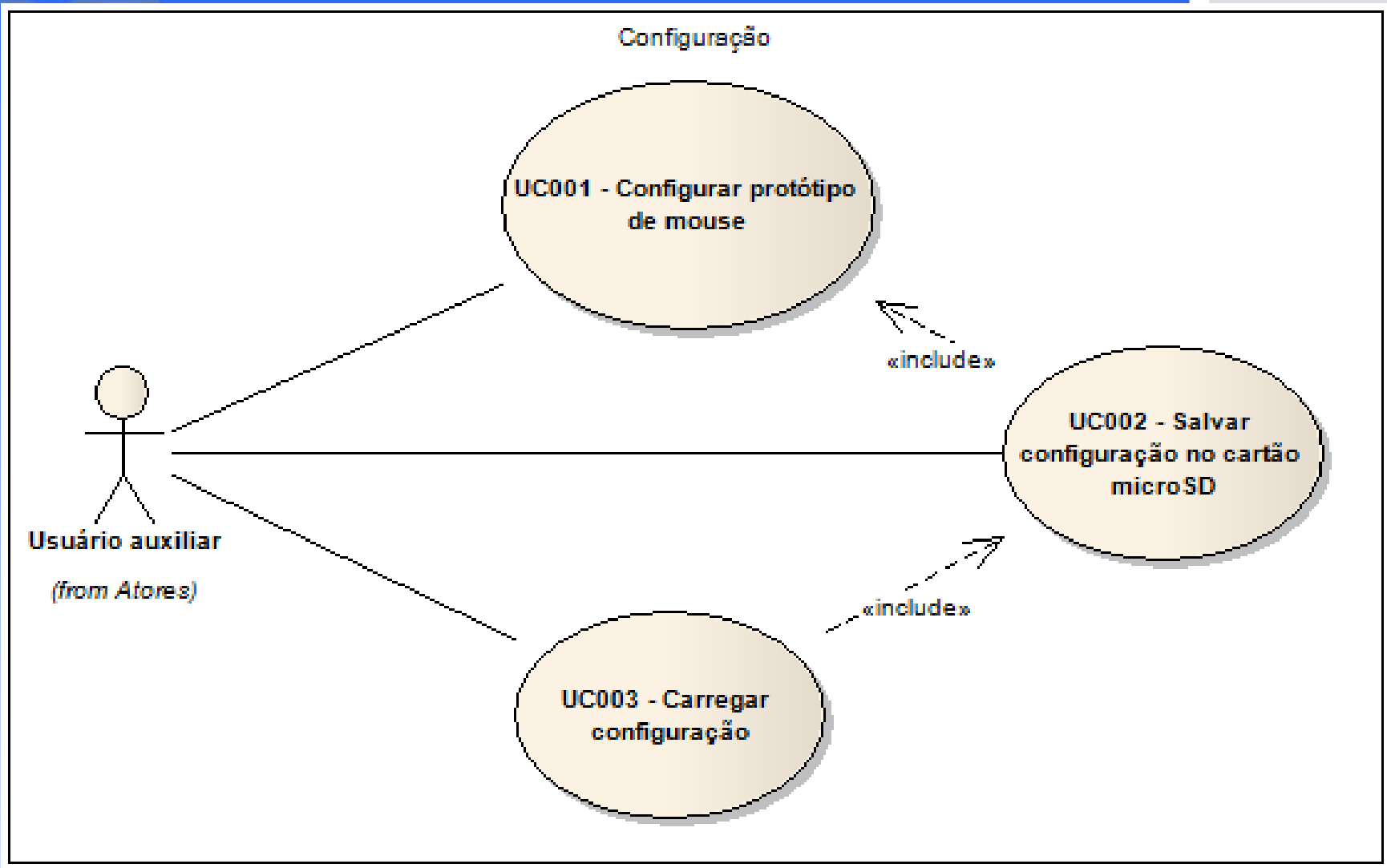
Desenvolvimento

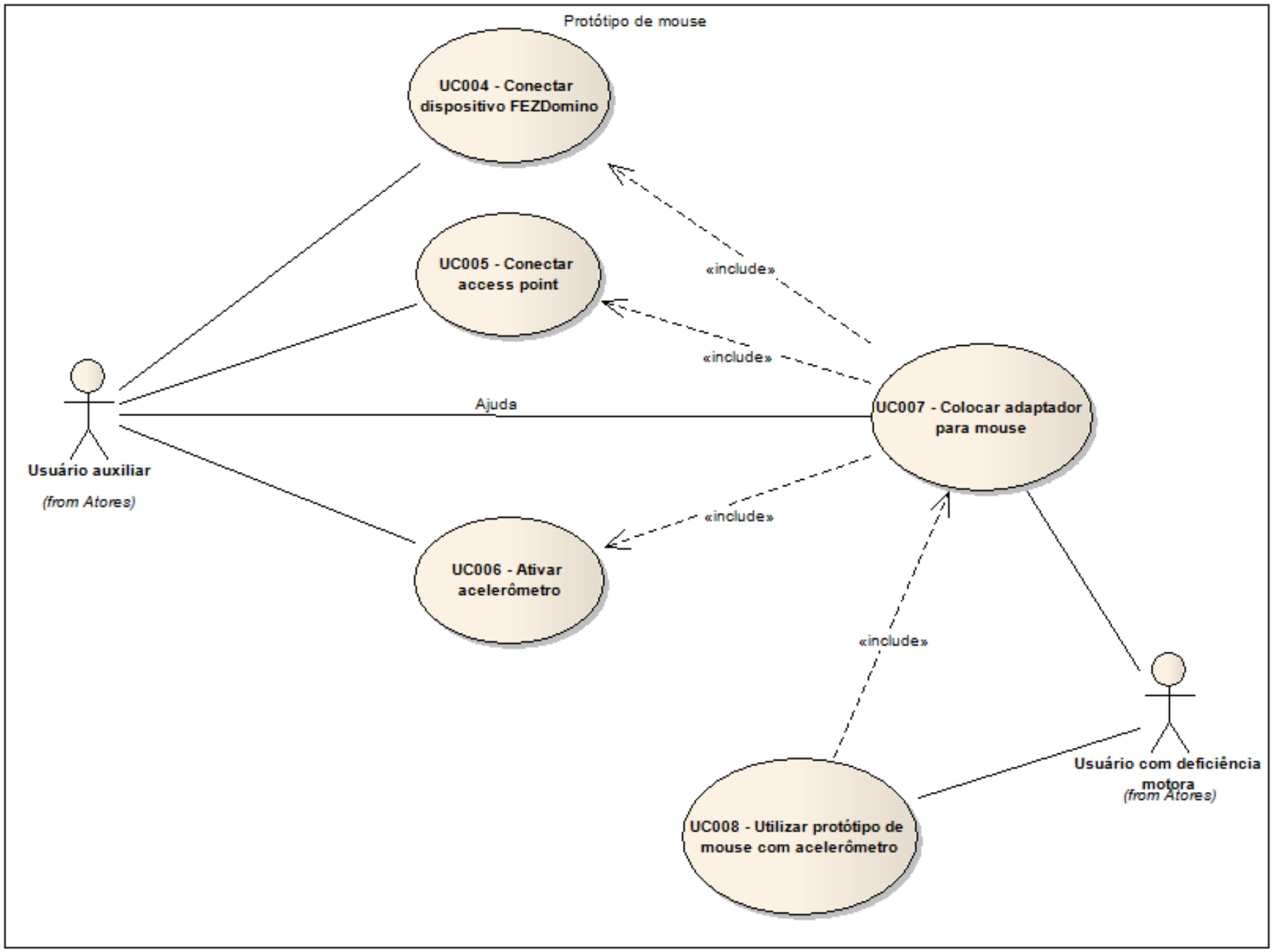
- Requisitos não funcionais
  - Utilizar Microsoft Visual Studio 2010.
  - Utilizar biblioteca .NET Micro Framework 4.1;
  - Utilizar micro-controlador FEZ Domino.
  - Permitir comunicação via conexão USB.
  - Ser multiplataforma.
  - Rodar em plataforma x86.

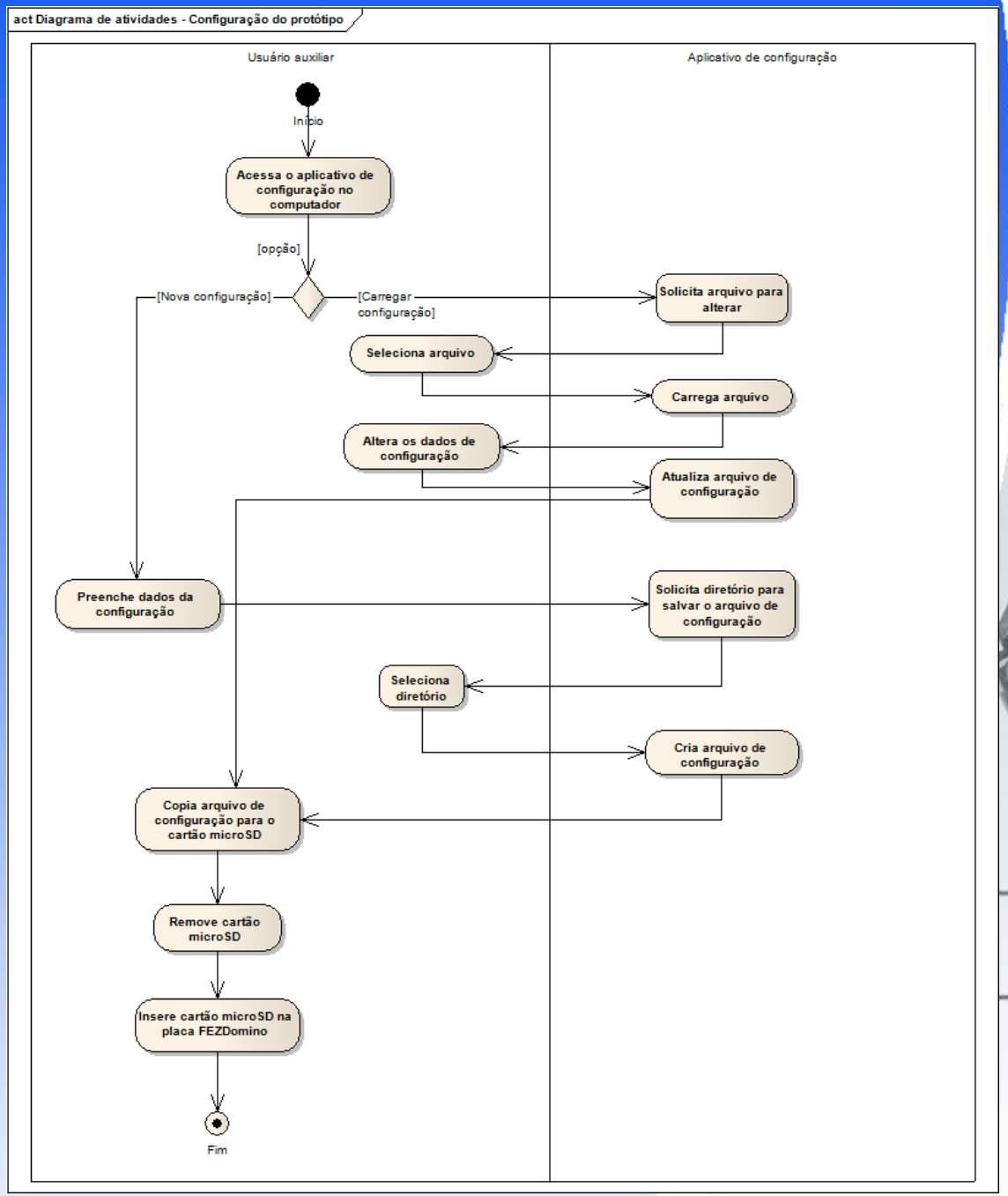
Protótipo

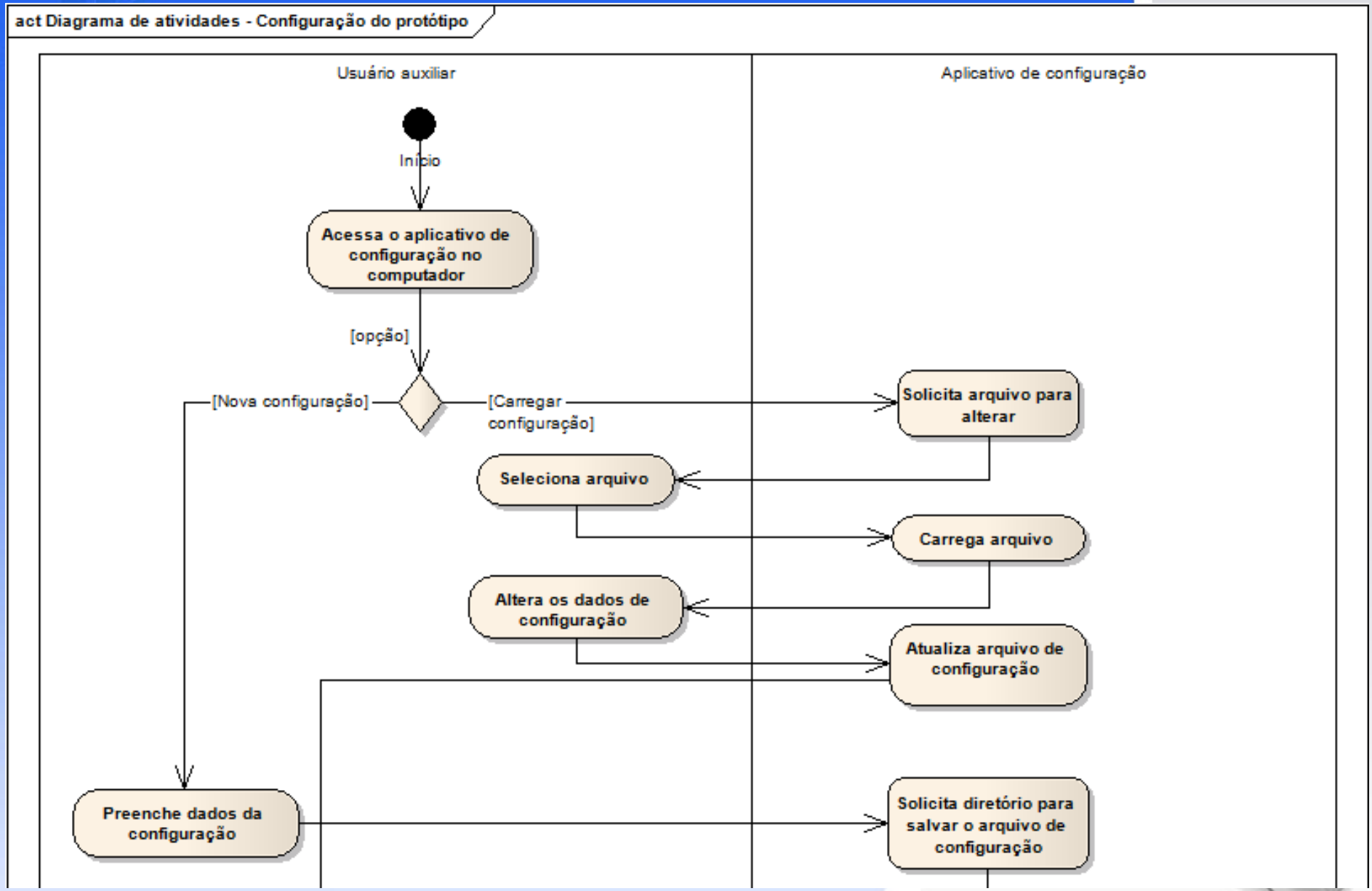

Aplicativo de configuração do protótipo de mouse - v1.0
Arquivo
Sobre
Movimento (Velocidade de deslocamento do ponteiro)
5
Lento
Rápido
Clique (Deslocamento para identificar um clique)
7
Pequeno
Grande
Salvar e fechar
Restaurar valores padrões

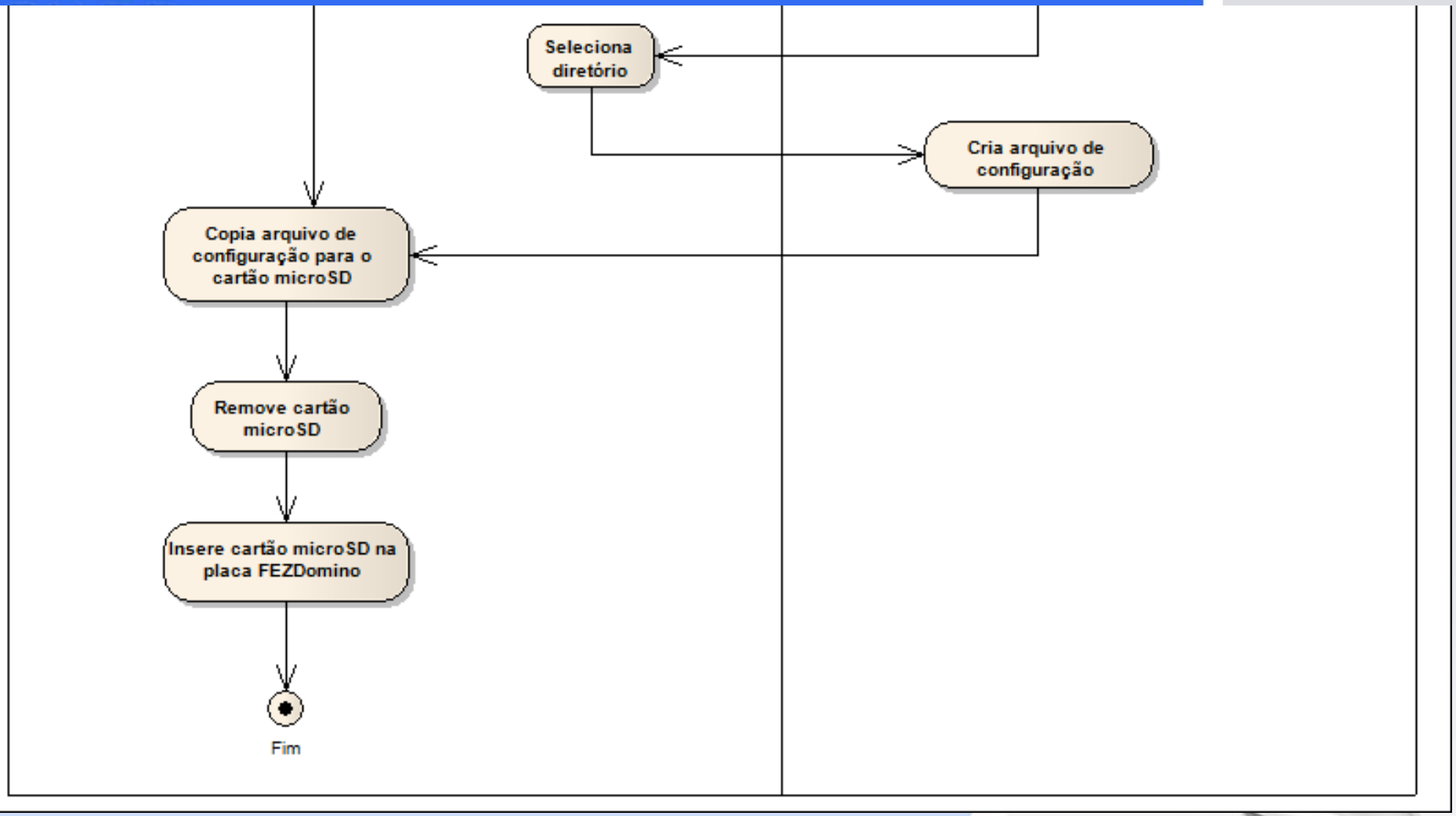
Especificação

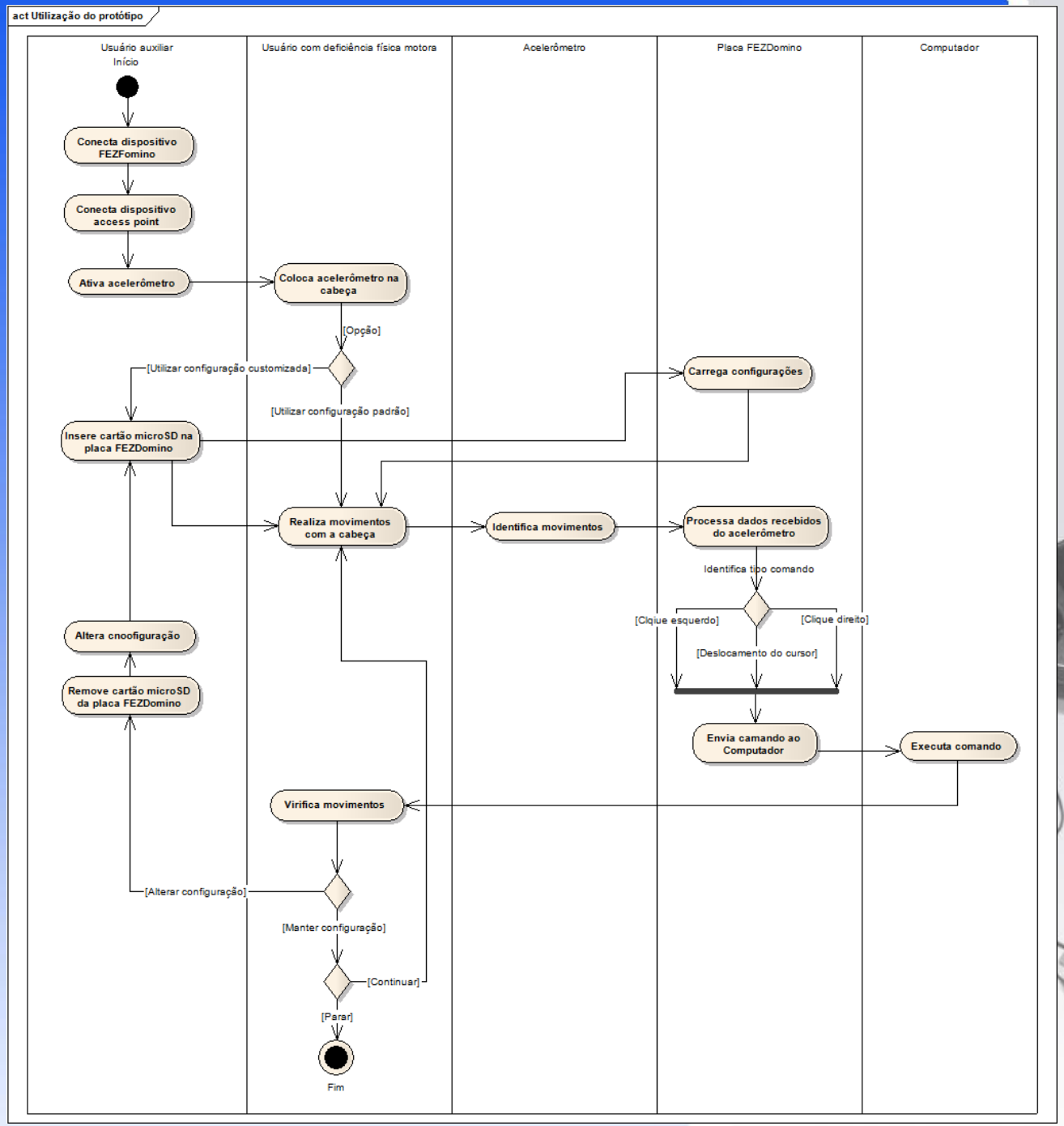

Configuração
UC001 - Configurar protótipo de mouse
«include»
UC002 - Salvar configuração no cartão microSD
Usuário auxiliar
(from Atores)
«include»
UC003 - Carregar configuração

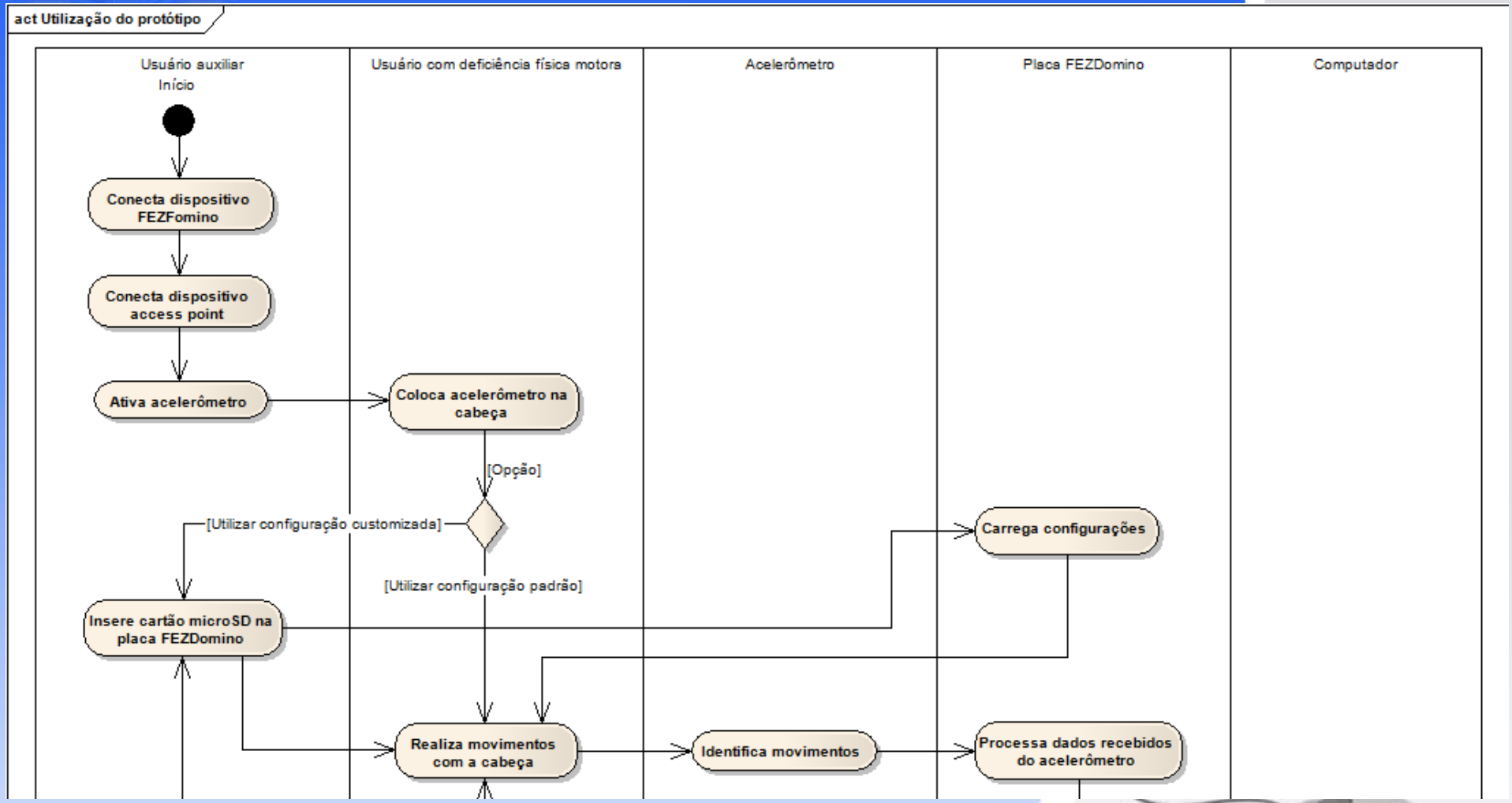

Protótipo de mouse
UC004 - Conectar dispositivo FEZDomino
UC005 - Conectar access point
«include»
«include»
Ajuda
UC007 - Colocar adaptador para mouse
Usuário auxiliar
(from Atores)
«include»
UC006 - Ativar acelerômetro
«include»
Usuário com deficiência motora
(from Atores)
UC008 - Utilizar protótipo de mouse com acelerômetro

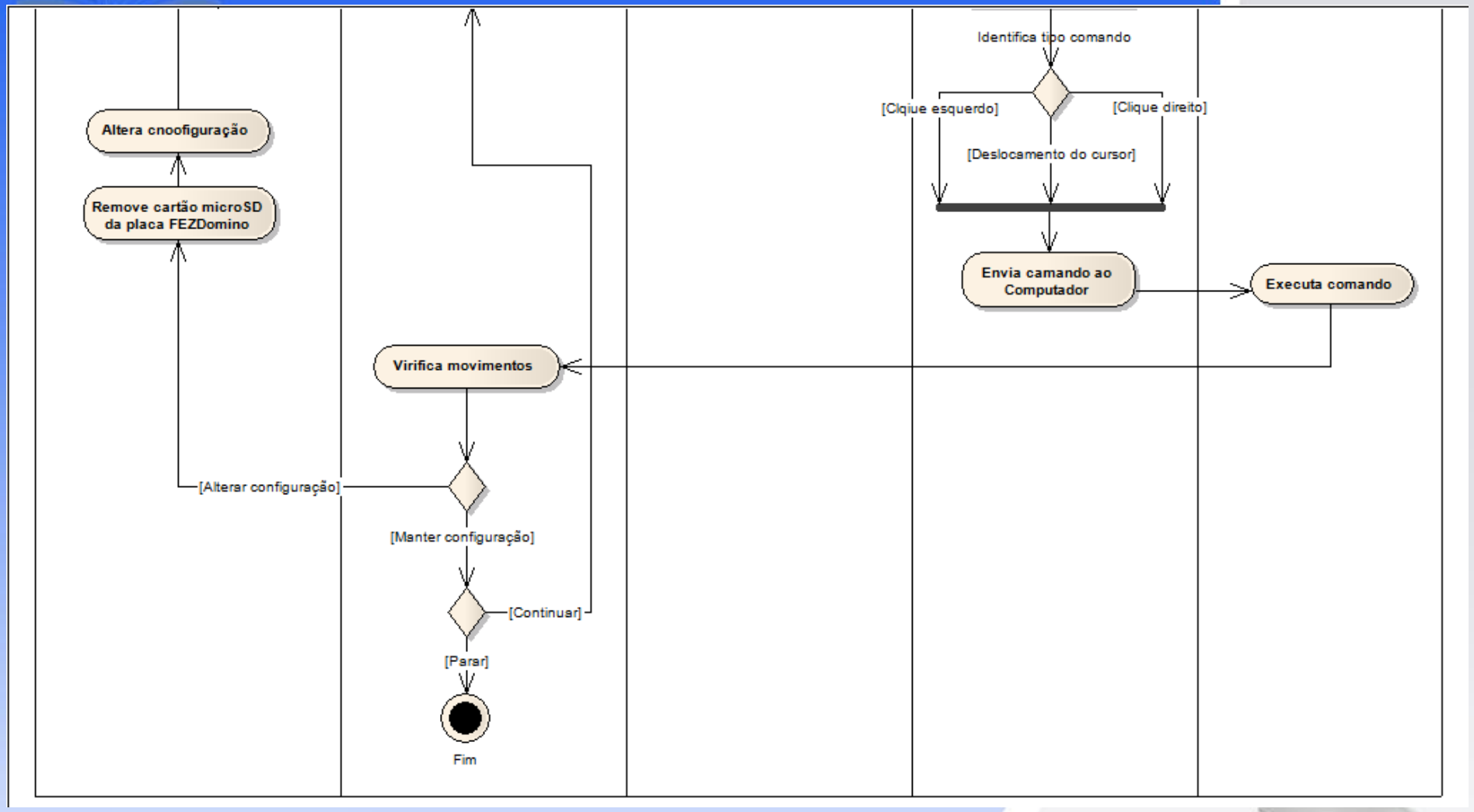

act Diagrama de atividades - Configuração do protótipo
Usuário auxiliar
Aplicativo de configuração
Início
Acessa o aplicativo de configuração no computador
[opção]
[Nova configuração]
[Carregar configuração]
Solicita arquivo para alterar
Seleciona arquivo
Carrega arquivo
Altera os dados de configuração
Atualiza arquivo de configuração
Preenche dados da configuração
Solicita diretório para salvar o arquivo de configuração
Seleciona diretório
Cria arquivo de configuração
Copia arquivo de configuração para o cartão microSD
Remove cartão microSD
Insere cartão microSD na placa FEZDomino
Fim


act Diagrama de atividades - Configuração do protótipo
Usuário auxiliar
Aplicativo de configuração
Início
Acessa o aplicativo de configuração no computador
[opção]
[Nova configuração]
[Carregar configuração]
Solicita arquivo para alterar
Seleciona arquivo
Carrega arquivo
Altera os dados de configuração
Atualiza arquivo de configuração
Preenche dados da configuração
Solicita diretório para salvar o arquivo de configuração


Seleciona diretório
Cria arquivo de configuração
Copia arquivo de configuração para o cartão microSD
Remove cartão microSD
Insere cartão microSD na placa FEZDomino
Fim


act Utilização do protótipo
Usuário auxiliar
Usuário com deficiência física motora
Acelerômetro
Placa FEZDomino
Computador
Início
Conecta dispositivo FEZFomino
Conecta dispositivo access point
Ativa acelerômetro
Coloca acelerômetro na cabeça
[Opção]
[Utilizar configuração customizada]
[Utilizar configuração padrão]
Carrega configurações
Insere cartão microSD na placa FEZDomino
Realiza movimentos com a cabeça
Identifica movimentos
Processa dados recebidos do acelerômetro
Identifica tipo comando
[Clique esquerdo]
[Clique direito]
[Deslocamento do cursor]
Altera cnoofiguração
Remove cartão microSD da placa FEZDomino
Envia camando ao Computador
Executa comando
Virifica movimentos
[Alterar configuração]
[Manter configuração]
[Continuar]
[Parar]
Fim


act Utilização do protótipo
Usuário auxiliar
Usuário com deficiência física motora
Acelerômetro
Placa FEZDomino
Computador
Início
Conecta dispositivo FEZFomino
Conecta dispositivo access point
Ativa acelerômetro
Coloca acelerômetro na cabeça
[Opção]
[Utilizar configuração customizada]
[Utilizar configuração padrão]
Carrega configurações
Insere cartão microSD na placa FEZDomino
Realiza movimentos com a cabeça
Identifica movimentos
Processa dados recebidos do acelerômetro


Identifica tipo comando
[Clqiue esquerdo]
[Clique direito]
[Deslocamento do cursor]
Envia camando ao Computador
Executa comando
Altera cnoofiguração
Remove cartão microSD da placa FEZDomino
Virifica movimentos
[Alterar configuração]
[Manter configuração]
[Continuar]
[Parar]
Fim

Implementação

- Preparação do hardware
  - Aquisição dos componentes.
- Aplicativo de configuração
  - Java.
  - IDE Netbeans.
  - Interface de configuração multiplataforma

Implementação

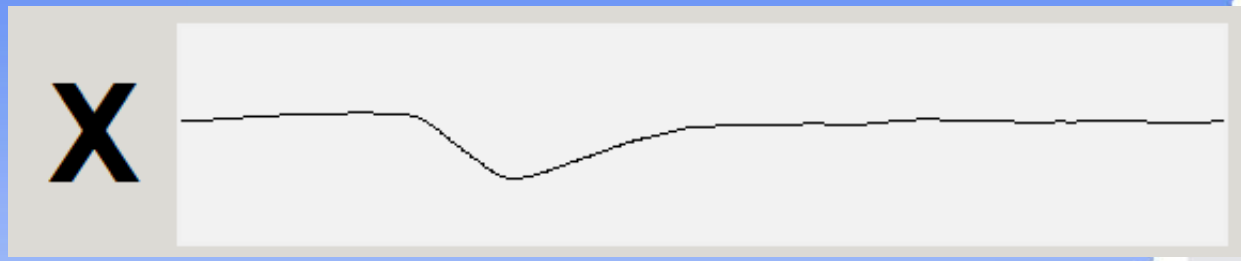
- Protótipo de mouse
  - C# .Net 4.0.
  - Visual Studio 2010.
  - .Net Micro Framework 4.1.

Protótipo de mouse

- Captura dos movimentos do acelerômetro.
- Leitura do cartão micro SD.
- Controle de velocidade do ponteiro.
  - Deslocamento * (Velocidade/ valor heurístico)

Protótipo de mouse

- Identificação de cliques através de acelerações
  - Texas Instruments eZ430-Chronos Control Center 1.1



- Estabilidade do ponteiro durante clique

Rotinas implementadas

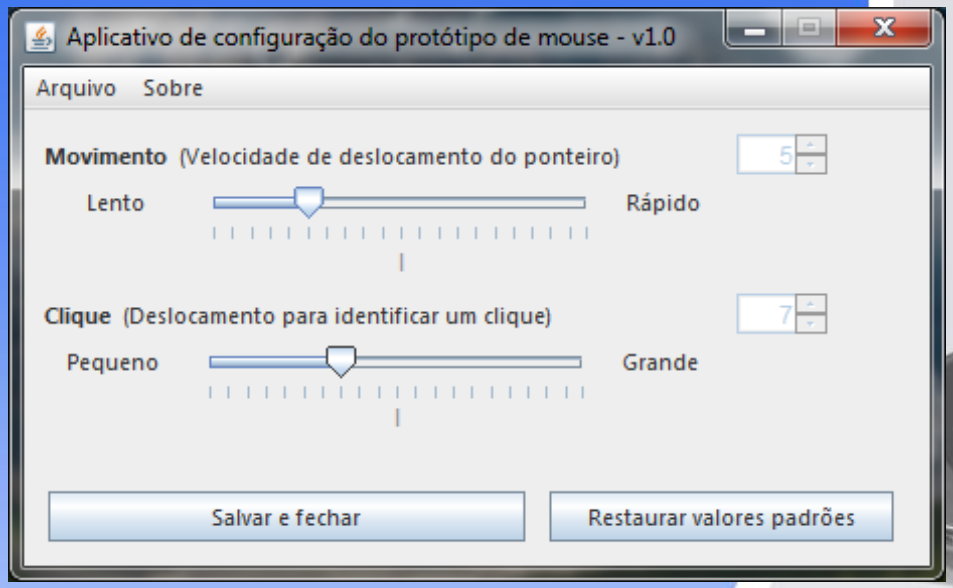
```
// Método utilizado para identificar cliques através de
// acelerações na inclinação do acelerômetro
private eTipoClique doVerificaClique(Deslocamento d, eTipoEixo eixo)
{
    int valorEixoAtual = 0;
    int valorEixoAnterior = 0;

    // Somente verifica o clique se já houve algum deslocamento
    if (p_UltimoDeslocamento != null)
    {
        Busca valor do eixo

        // Verifica se o mouse foi estabilisado depois do último clique efetuado
        if (p_EstabilizouAposClique || (p_UltimoClique != null && p_UltimoClique.TipoClique == eTipoClique.Esquerdo))
        {
            // Verifica se houve aceleração no eixo
            if (isDeslocamentoUltrapassaIndice(valorEixoAtual, valorEixoAnterior, p_MouseConfig.SensibilidadeClique))
            {
                eTipoClique clique = eTipoClique.Nenhum;
                if ((valorEixoAtual > const_NivelEstabilidadeEixoZ) || (valorEixoAtual < 0))
                    clique = eixo == p_MouseConfig.EixoCliqueEsquerdo ? eTipoClique.Esquerdo : eTipoClique.Direito;

                if ((p_UltimoClique) != null)
                {
                    // Verifica se houve um intervalo entre o clique atual e o último clique
                    int intervalo = p_UltimoClique.TipoClique == eTipoClique.Esquerdo && clique == eTipoClique.Esquerdo ?
                        p_MouseConfig.IntervaloMinimoEntreCliqueEsquerdo : p_MouseConfig.IntervaloMinimoEntreClique;

                    // Verifica se atingiu o intervalo mínimo estre cliques
                    if (p_UltimoClique.HoraClique.AddMilliseconds(intervalo) > DateTime.Now)
                        clique = eTipoClique.Nenhum;
                }

                Se houve clique, seta as variáveis de controle
                return clique;
            }
        }
    }
    if (!p_EstabilizouAposClique)
    {
        if (isDeslocamentoDentroIndice(valorEixoAtual, p_MouseConfig.SensibilidadeDeteccaoMovimento + 1))
            p_EstabilizouAposClique = true;
        else if(p_UltimoClique != null && p_UltimoClique.HoraClique.AddMilliseconds(3000) > DateTime.Now)
            p_EstabilizouAposClique = true;
    }
    return eTipoClique.Nenhum;
}
```

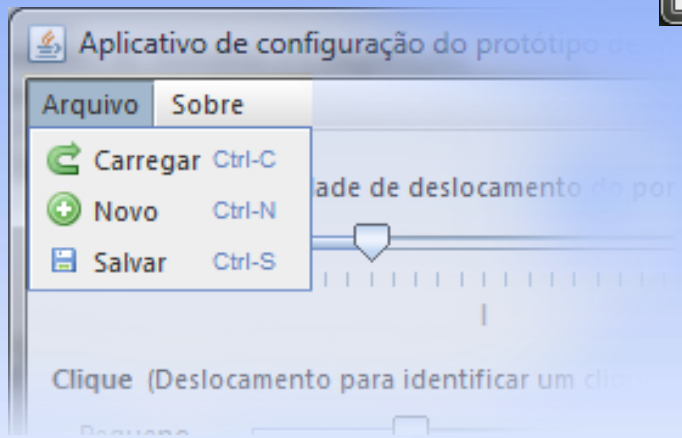
```
// Método utilizado para aplicar a fórmula de controle de
// velocidade para os eixos X e Y
private void doVerificarSensibilidadeMovimento(Deslocamento d)
{
    d.Eixo_X = doAplicarVelocidadeMovimento(d.Eixo_X);
    d.Eixo_Y = doAplicarVelocidadeMovimento(d.Eixo_Y);
}

private int doAplicarVelocidadeMovimento(int deslocamentoEixo)
{
    // Fórmula = deslocamento * (velocidade/indice)
    double velocidade = double.Parse(p_MouseConfig.VelocidadeMovimento.ToString());
    double deslocamento = double.Parse(deslocamentoEixo.ToString());
    double novaVelocidade = deslocamento * (velocidade / 5);

    return int.Parse(System.Math.Round(novaVelocidade).ToString()); ;
}
```

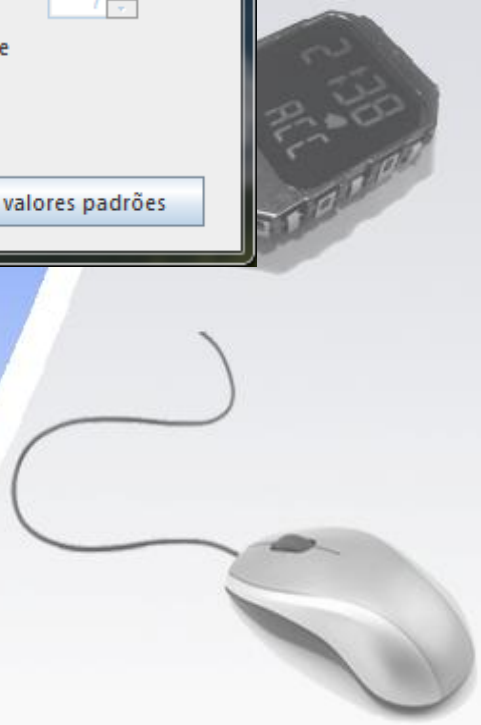
```
public class Mouse : IDisposable
{
    private static byte Flag = 0;
    [...]
    private void doMover(Deslocamento d)
    {
        try
        {
            // X = Movimento Vertical
            // Y = Movimento horizontal

            // Verificar se o dispositivo está conectado ao PC
            if (USBClientController.GetState() == USBClientController.State.Running)
                p_Mouse.SendData(d.Eixo_Y, d.Eixo_X*-1, 0, USBC_Mouse.Buttons.BUTTON_NONE)
        }
        catch (Exception err)
        {
            throw new Exception("Erro ao mover mouse! \r\nMensagem: " + err.Message);
        }
    }
    [...]
    public void Dispose()...
}
```

Operacionalidade da implementação

Usuário auxiliar

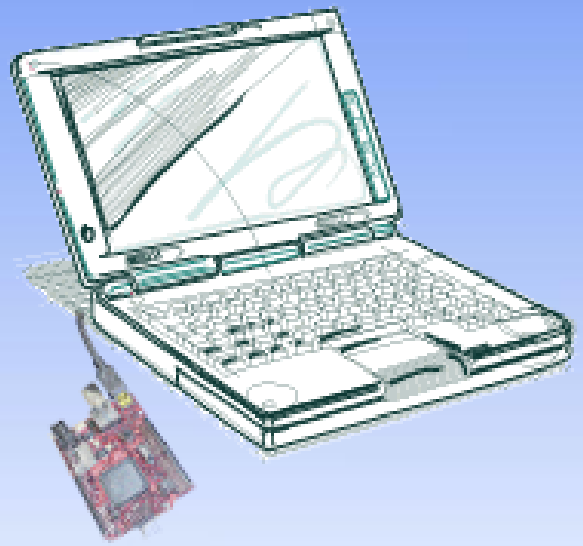






Operacionalidade da implementação

Usuário auxiliar



Usuário com deficiência motora

Resultados e discussão

- Ótima usabilidade.
- Multiplataforma.
- Fácil configuração.
- Alto consumo de bateria.
- Falta do clique seleção.
- Dificuldades encontradas.

Trabalhos relacionados

TRABALHOS CORRELATOS					
Funcionalidade	Tongue Drive	Mouse visual Bradesco	Capacete Emotiv EPOC	Protótipo Jennrich (2010)	**Protótipo versão 2 (2011)**
Independente de aplicativos	--	--	--	--	**X**
Multiplataforma	--	--	X	--	**X**
Movimenta cursor	X	X	X	X	**X**
Clique direito	--	--	--	X	**X**
Clique esquerdo	X	X	--	X	**X**
Clique seleção	--	--	--	X	**--**
Clique duplo	X	--	--	X	**X**
Comunicação sem fio	X	X	X	--	**X**

Conclusão

- Solução para mouse adaptados.
- Comercialização.

Extensões

- Clique seleção.
- Redesenhar Hardware.
- Testar em uma entidade especializada.

Demonstração

Obrigado