



# Protótipo de uma aplicação rica de internet para monitoramento de vídeo através de streaming e Silverlight

Thiago da Silva Negherbon – Acadêmico  
Roosevelt dos Santos Júnior – Orientador

# Roteiro

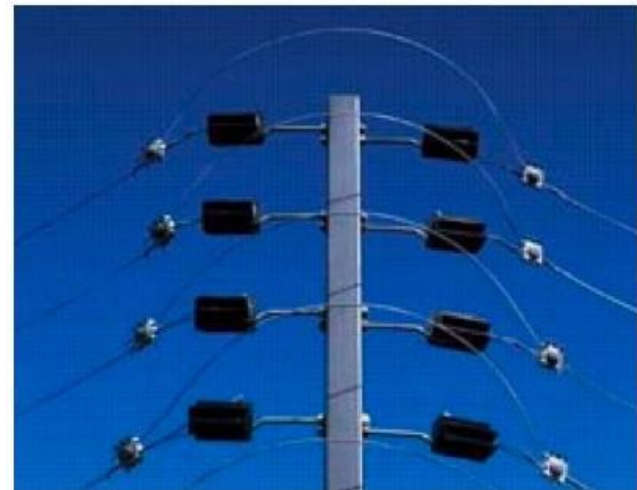


- Introdução
  - Objetivos do trabalho
- Fundamentação teórica
  - RIA, Silverlight, XAML, *streaming*, compressão, DirectShow, trabalhos correlatos
- Desenvolvimento do protótipo
  - Requisitos principais, especificação, implementação, resultados e discussão
- Conclusão
  - Extensões

# Introdução



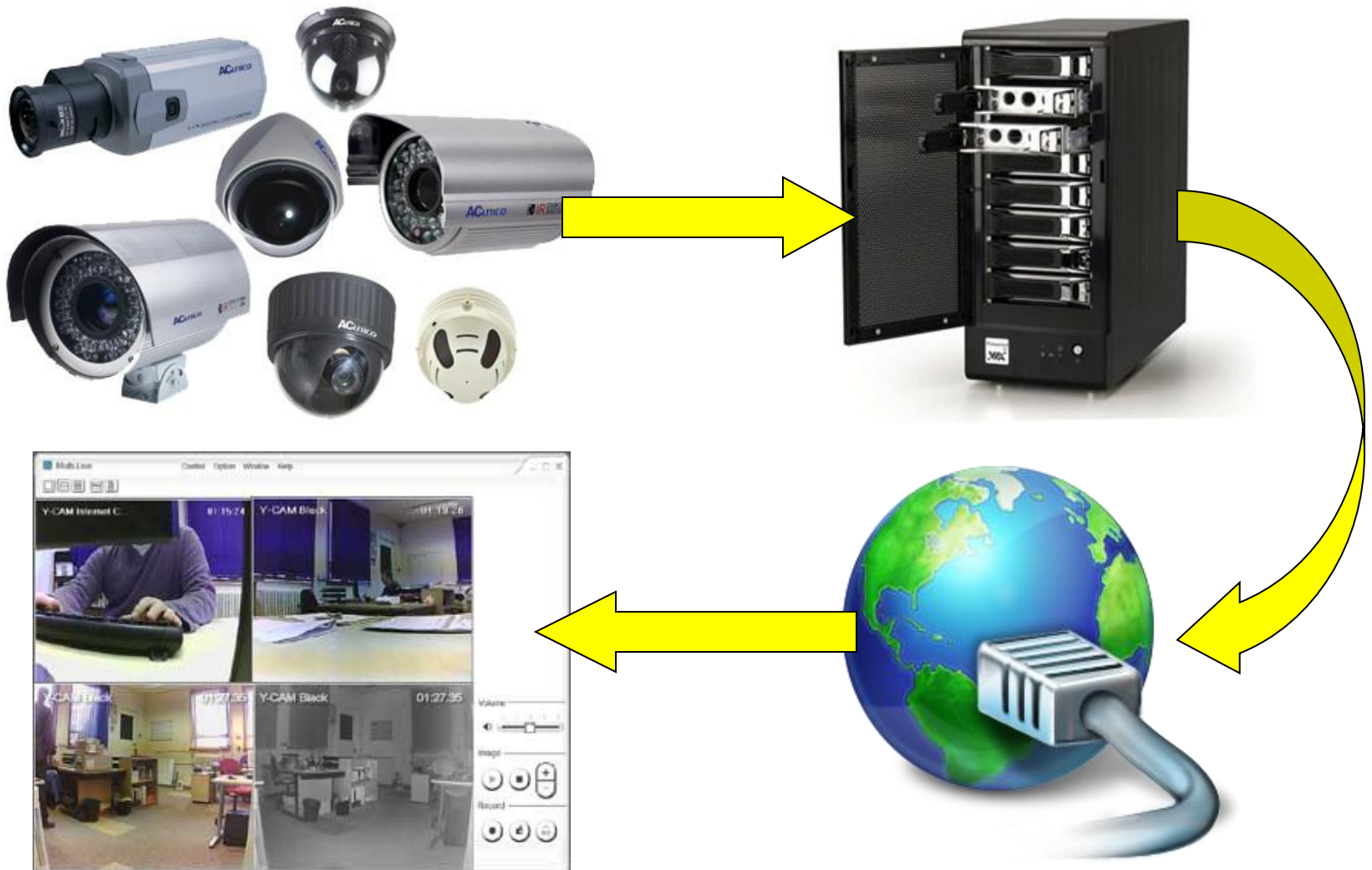
# Introdução



# Introdução



# Objetivos do trabalho



# Fundamentação teórica

# Rich Internet Application



[Search the repository](#) | [Browse for research](#) | [About this repository](#) | [Repository policies](#) | [FAQ](#) | [Contact](#)

## **Patterns in HCI: A critical review**

Full text available: [Patterns in HCI: A critical review](#)

<b>Abstract:</b>	This article presents a critical review of patterns and pattern languages in human-computer interaction (HCI). In recent years, patterns and pattern languages have received considerable attention in HCI for their potential as a means for developing and communicating information and knowledge to support good design. This review examines the background to patterns and pattern languages in HCI, and seeks to locate pattern languages in relation to other approaches to interaction design. The review explores four key issues: What is a pattern? What is a pattern language? How are patterns and pattern languages used? and How are values reflected in the pattern-based approach to design? Following on from the review, a future research agenda is proposed for patterns and pattern languages in HCI.
<b>ISSN:</b>	1532-7051
<b>DOI:</b>	10.1207/s15327051hci2101_3
<b>Reference:</b>	Dearden, A., Finlay, J. (2006) Patterns in HCI: A critical review. <b>Human Computer Interaction Journal</b> , 21 (1) March, pp.49-102.
<b>Item type:</b>	Journal Item
<b>Copyright:</b>	Taylor & Francis (2006) Uploaded in accordance with the publisher's self archiving policy.
<b>Published status:</b>	Published
<b>Published URL:</b>	<a href="http://www.informaworld.com/smpp/content~content=a795059926~db=all">http://www.informaworld.com/smpp/content~content=a795059926~db=all</a>
<b>Refereed status:</b>	Refereed
<b>Classification:</b>	T - Technology
<b>Deposited on:</b>	2nd July 2009, 8:26
<b>Author(s):</b>	<a href="#">Dearden, Andy</a> , <a href="#">Finlay, Janet</a>
<b>Publisher:</b>	<a href="#">Taylor &amp; Francis</a>
<b>Deposited by:</b>	<a href="#">Sheppard, Nick</a>
<b>Subject(s):</b>	<a href="#">Human-computer interaction</a> , <a href="#">Ergonomics design and management</a>

[Leeds Met Home](#)

[intraLibrary Login](#)



# Rich Internet Application



# Rich Internet Application



Microsoft®  
**Silverlight™**

# Silverlight



The screenshot shows a web application interface for 'Avee Talent Management'. At the top, there is a navigation bar with 'NetAdvantage' branding and links for 'COMMUNITY', 'BLOGS', and 'DOCUMENTATION'. Below this is a toolbar with various icons for navigation and actions. The main content area features a header with the 'Avee Talent Management' logo and a user login status 'Signed in as Sam Bulagira'. A 'CONTACTS' tab is active, displaying a list of contact names in a vertical stack. The primary contact, 'Ana Trujillo', is highlighted with a detailed view including a profile picture, contact information, a bar chart showing activity from January to November, and a 'SYSTEM' log of recent transactions. A 'CUSTOMER' log provides additional context on contact updates.

Home > Avee Talent Management

**AVEE TALENT MANAGEMENT**  
Avee uses the XamTileView to make contact management engaging. Maximize a tile to see client details. The XamTileView supports Silverlight DataBinding, enabling layouts like Avee via templating.

Signed in as Sam Bulagira [Help](#) [Sign Out](#)

CURRENT VIEW **CONTACTS**

**Ana Trujillo**  
Ana Trujillo Emparedados y helados  
phone: (5) 555-4729  
fax: (5) 555-3745  
atrujillo@example.com

**SYSTEM**

Last Payment:	10/14/2008
Last Shipment:	10/11/2008
Last Invoice:	10/5/2008
Last Order:	10/4/2008

**CUSTOMER**

**07 Aug 2008**  
Customer called to update contact info; previous contact was let go due to incompetencia.

**23 Sep 2008**  
Customer called back to update shipping information for order 10926.

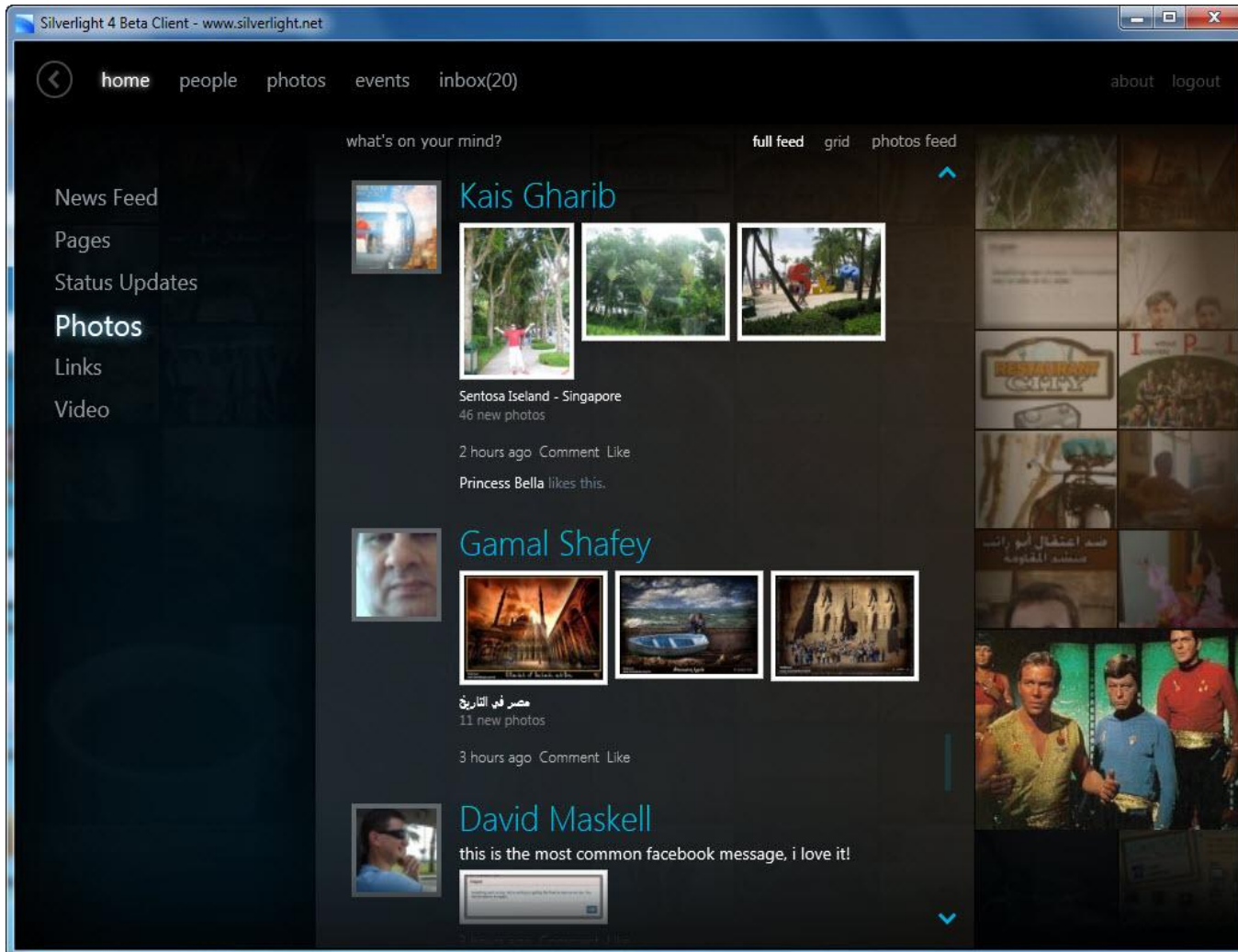
Bar Chart Data (Estimated):

Month	Value
Jan	1.0
Apr	1.5
May	3.0
Jun	2.5
Sept	1.5
Nov	0.5

Vertical Contact List:

- Maria Anders
- Antonio Moreno
- Thomas Hardy
- Christina Berglund
- Hanna Moos
- Elizabeth Citeaux
- Martín Sommer
- Laurence Lebihan
- Jon Hammond

# Silverlight



# Silverlight



## Media browser <sup>Beta</sup>

User: visitor  
Last connection:  
Level: 0  
[Config](#) [Fullscreen](#)

**Medias**

Type:  All  Image  Mp3  Streaming

Filter: File:  Folder:

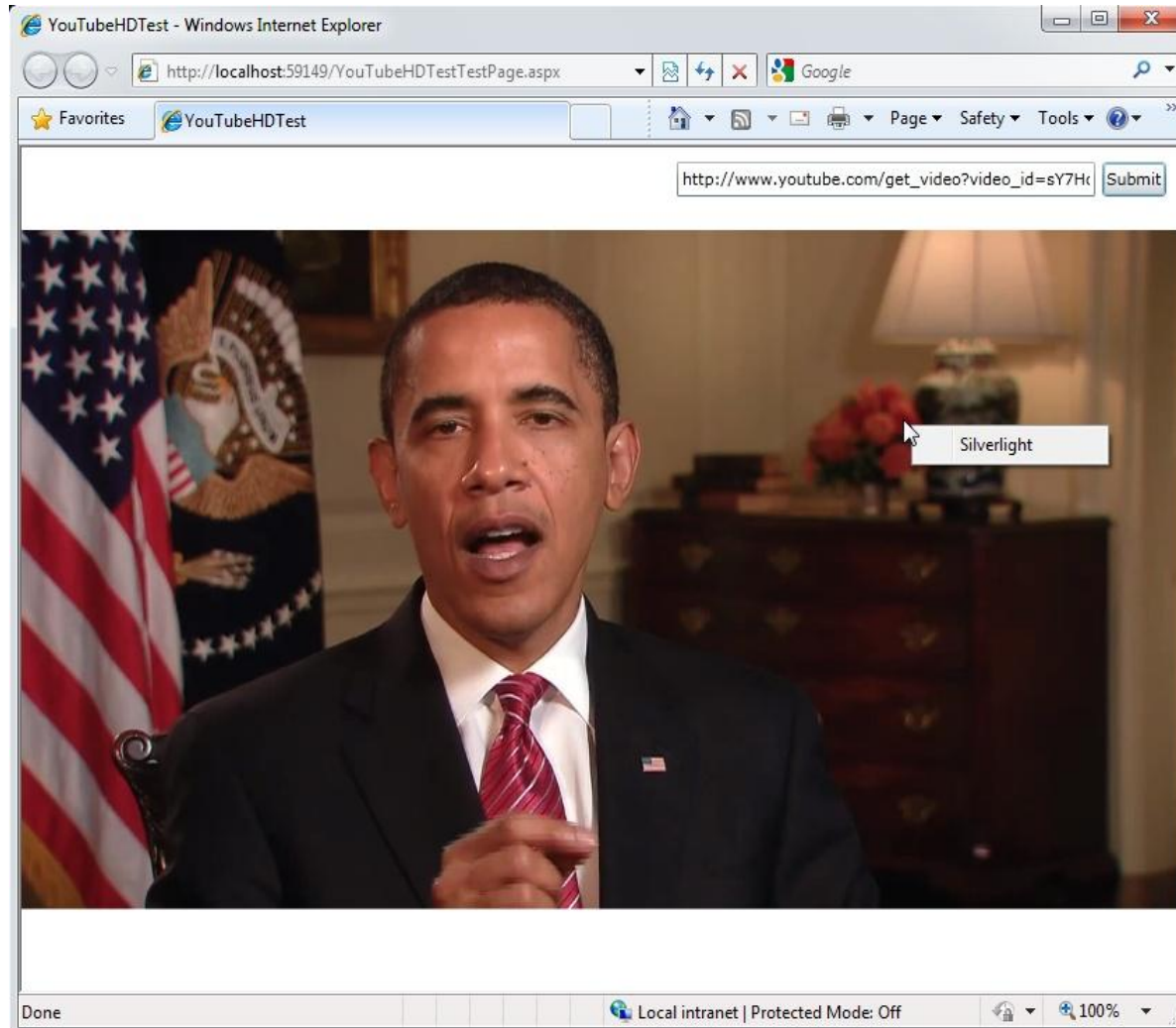
Current selection: [Refresh](#)

[Open](#) [Edit](#) [Delete](#) [Download](#)

Filename: ferrari\_aurea\_gt\_800.jpg  
Date added: 12/14/2008 4:02:09 PM  
Uploader: visitor2  
Description:

A grid of 25 thumbnails showing various images of cars and airplanes. The first row contains five airplane images. The second row contains five more airplane images. The third row contains five car images, including a red Ferrari and a blue car. The fourth row contains five car images, including a red Ferrari and a yellow car. The fifth row contains five car images, including a red Ferrari and a green car. The sixth row contains five car images, including a silver car and a black car.

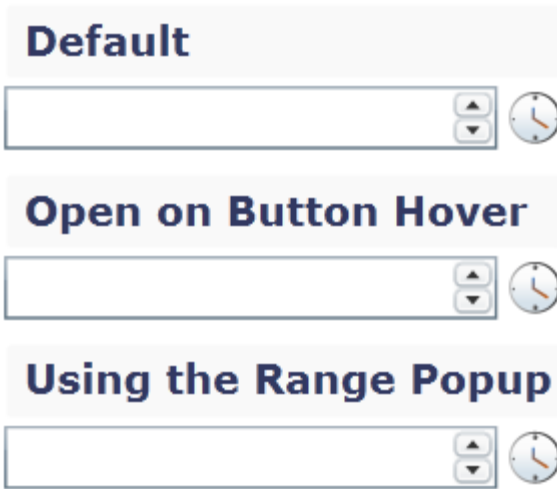
# Silverlight



# eXtensible Application Markup Language



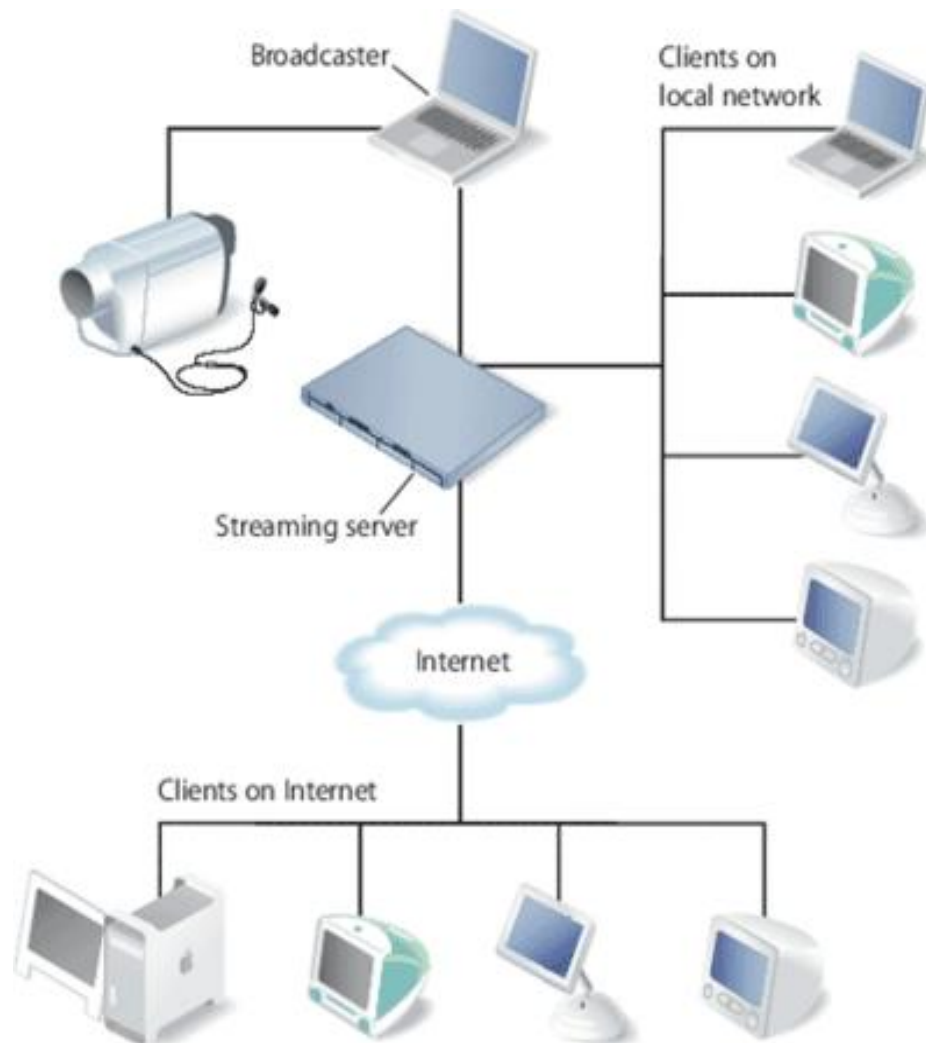
```
<UserControl ... >
  <StackPanel Background="White">
    <ContentControl ...
      <inputToolkit:TimePicker
      </inputToolkit:TimePicker>
    </ContentControl>
    <ContentControl ...
      Header}" />
      <inputToolkit:TimePicker
      </inputToolkit:TimePicker>
    </ContentControl>
    <ContentControl ...
      Header}" />
      <inputToolkit:TimePicker
        <inputToolkit:TimePicker.Popup
      </inputToolkit:TimePicker>
    </ContentControl>
  </StackPanel>
</UserControl>
```



The image shows three examples of the TimePicker control in a Windows application. Each example is presented in a separate window with a title bar. The first window is titled "Default" and shows a standard TimePicker control with a white background and a clock icon. The second window is titled "Open on Button Hover" and shows a TimePicker control with a white background and a clock icon, where the popup is open. The third window is titled "Using the Range Popup" and shows a TimePicker control with a white background and a clock icon, where the popup is open and displays a range of time values.

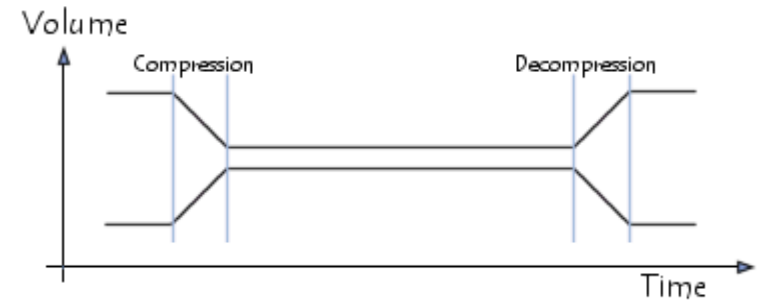
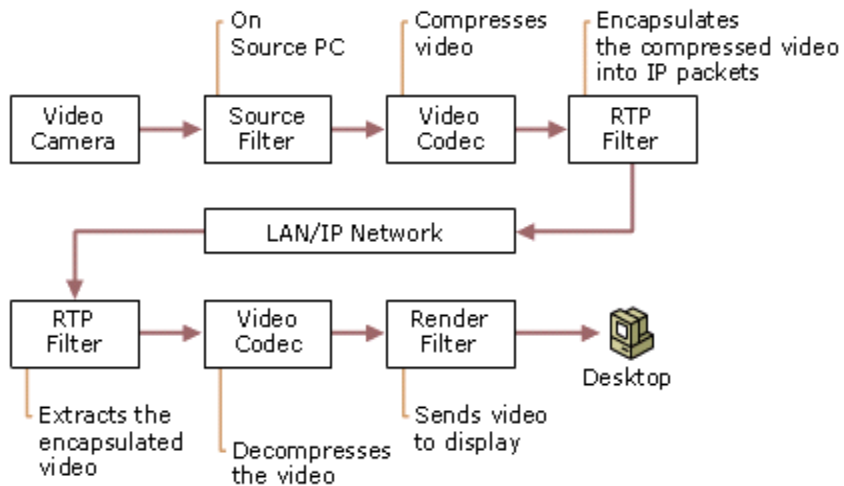
```
cResource Header}" />
Style="{StaticResource
"/>
Style="{StaticResource
:Popup />
```

# Streaming





# Compressão

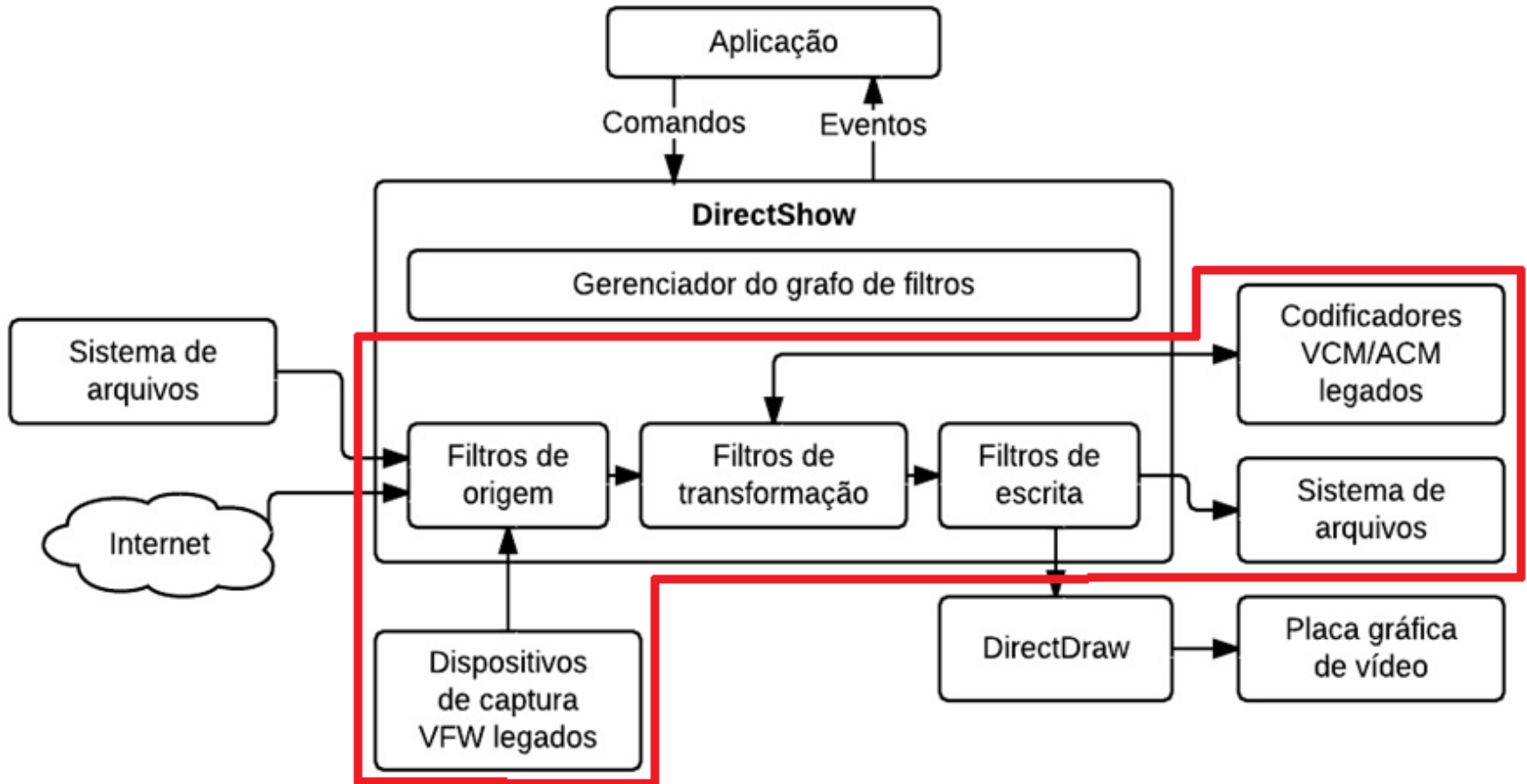


# Compressão

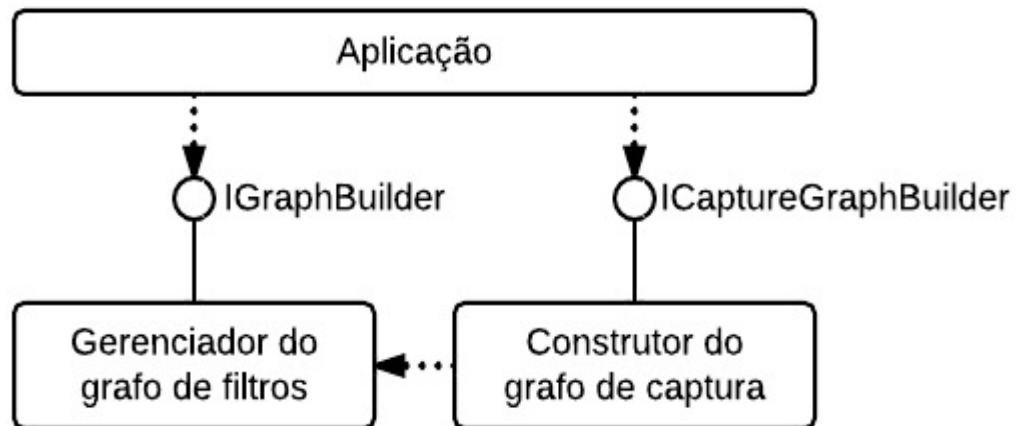


Aplicação	Taxa de dados	
	Descomprimido	Comprimido
Voz	64 kbps	2 – 4 kbps
Vídeo conferência (15 fps) tamanho do quadro 352 x 240, 24 <i>bits/pixel</i>	30,41 mbps	64 – 768 kbps
Transmissão de vídeo (30 fps) tamanho do quadro 720 x 480, 24 <i>bits/pixel</i>	248,83 mbps	3 – 8 mbps
Televisão de alta definição (59,94 fps) tamanho do quadro 1280 x 720, 24 <i>bits/pixel</i>	1,33 gbps	20 mbps

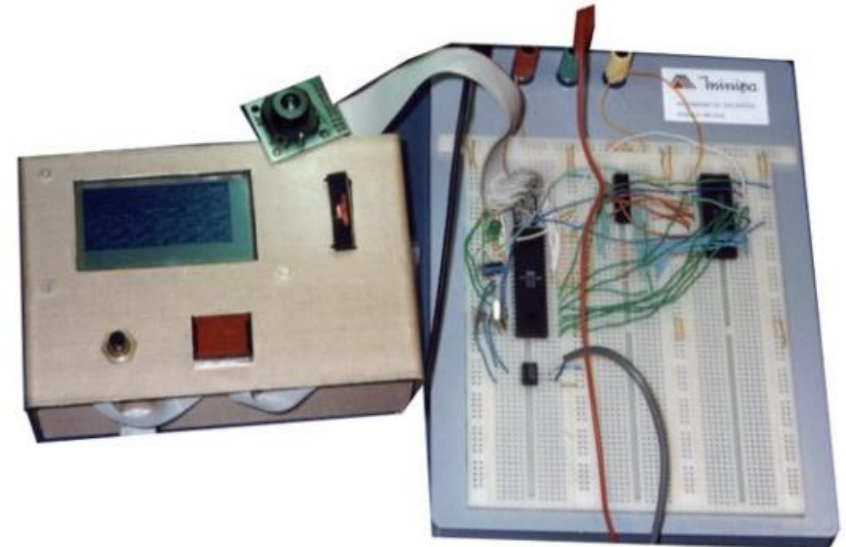
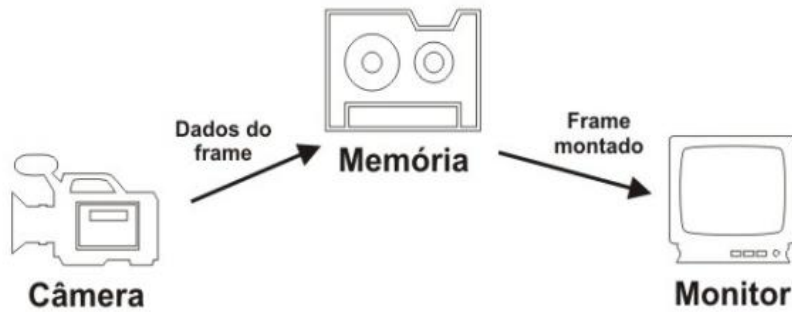
# DirectShow



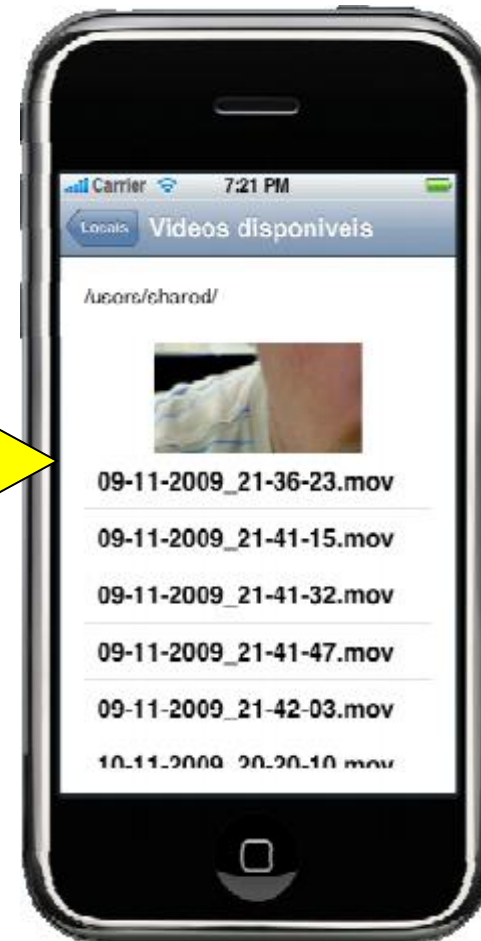
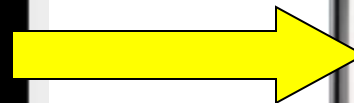
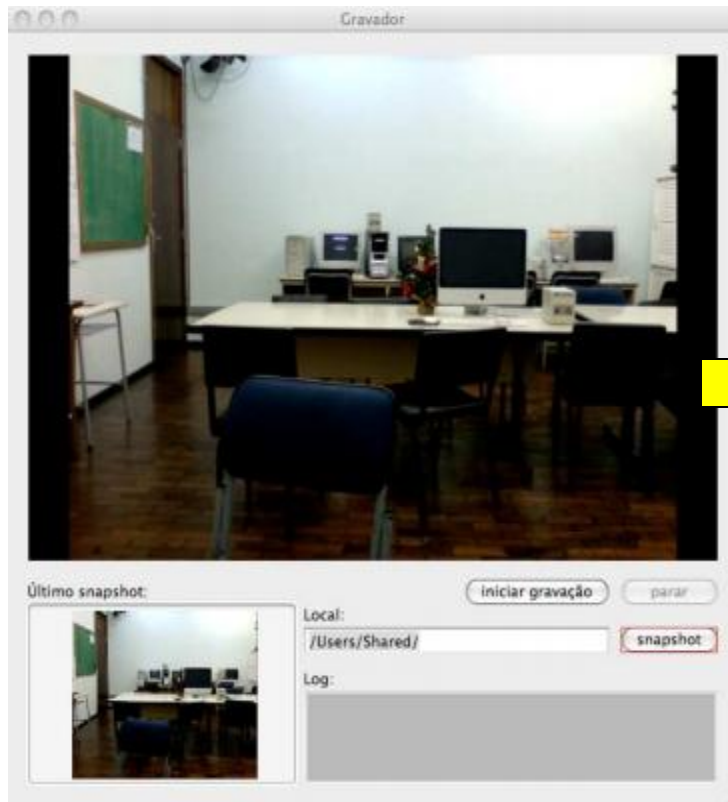
# DirectShow



# Trabalho de Mereço Neto (2004)



# Trabalho de Carlassara (2009)



# Desenvolvimento do protótipo

# Requisitos funcionais



- RF01: A aplicação cliente deve disponibilizar um painel que ocupe o maior espaço possível da janela do navegador onde o usuário poderá interagir.



# Requisitos funcionais



- RF02: A aplicação cliente deve disponibilizar um quadro para cada *streaming* de câmera que estiver recebendo.

# Requisitos funcionais



- RF03: A aplicação cliente deve possuir um menu onde podem ser escolhidas as câmeras escondidas que se deseja restaurar.

# Requisitos funcionais



- RF04: A aplicação cliente deve permitir redimensionamento e movimentação livres (limitado pelo tamanho do painel) de cada quadro de câmera.

# Requisitos funcionais



- RF05: A aplicação cliente deve permitir fechar cada quadro de câmera.

# Requisitos funcionais



- RF06: A aplicação cliente deve permitir parar e continuar a exibição das imagens de cada janela de câmera.

# Requisitos funcionais



- RF07: A aplicação servidora deve ler a entrada de cada câmera conectada ao computador.

# Requisitos funcionais



- RF08: A aplicação servidora deve processar os dados e criar o *streaming* de vídeo de cada câmera.

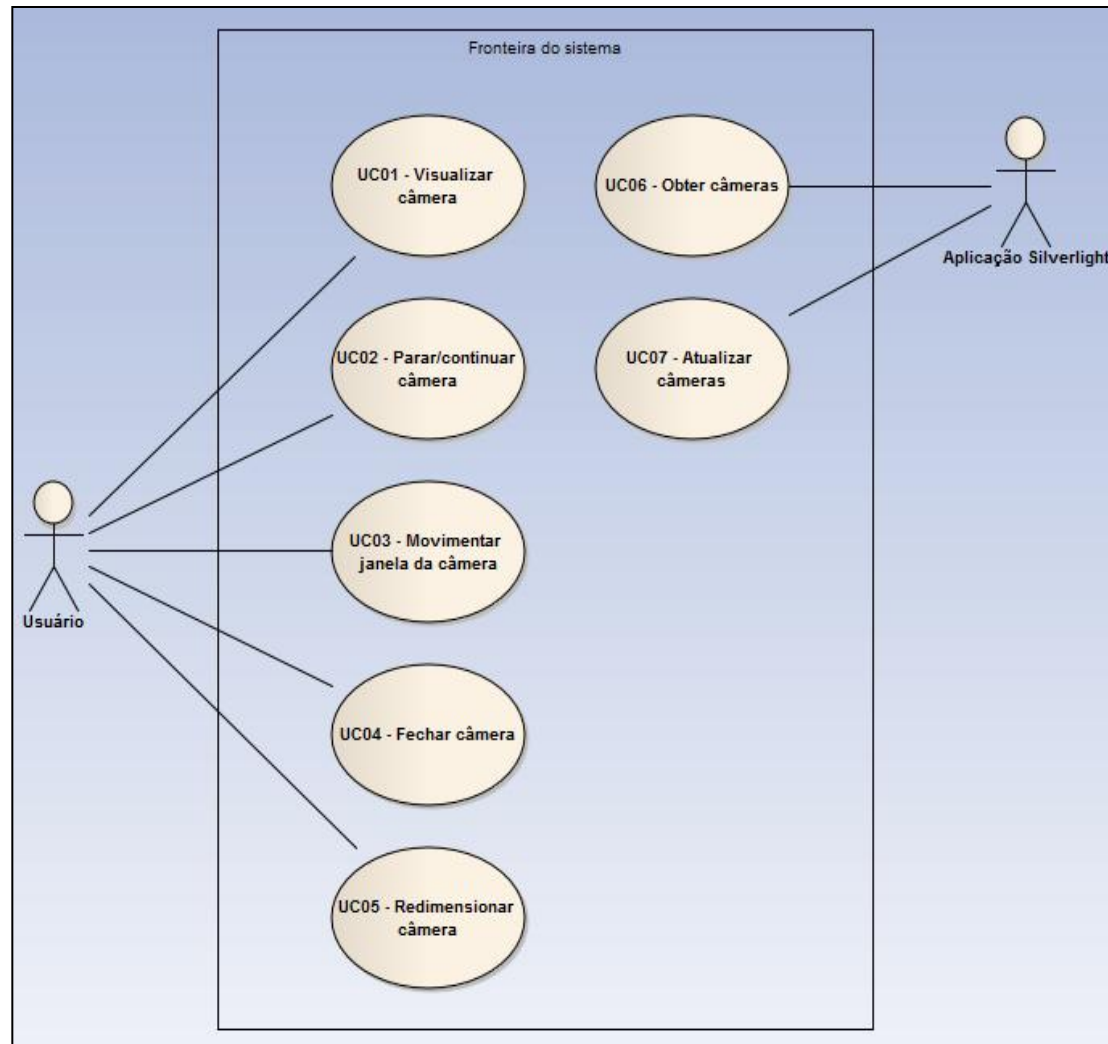
# Requisitos funcionais



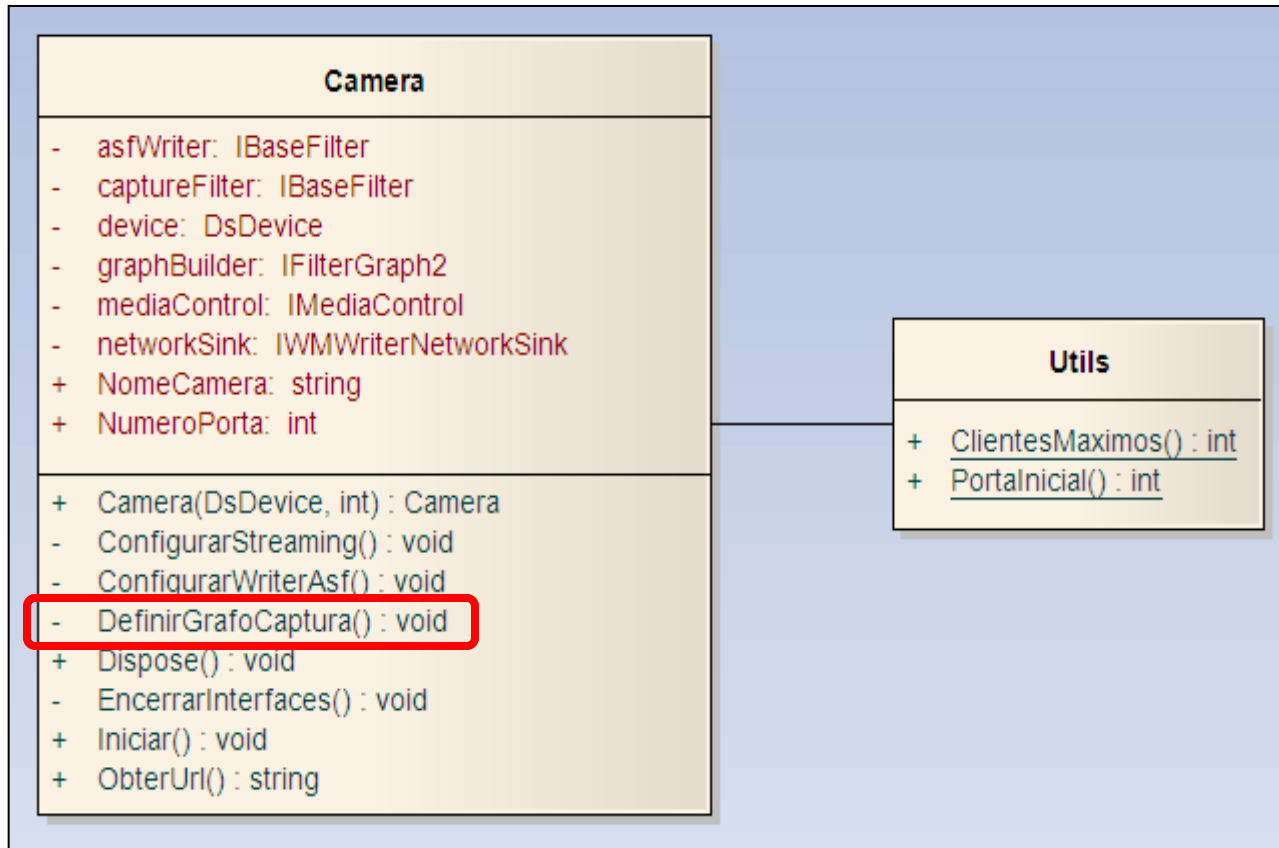
- RF09: A aplicação servidora deve disponibilizar os *streamings* para leitura na internet.



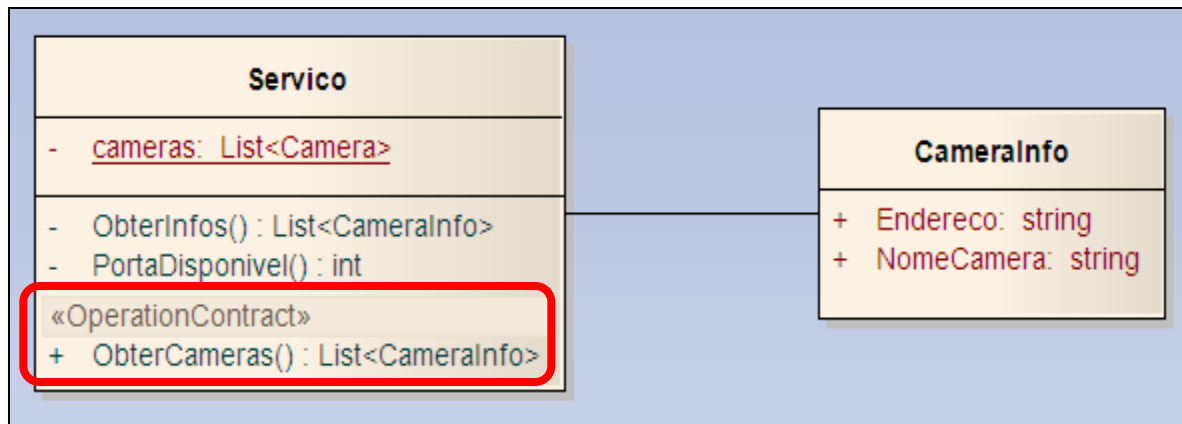
# Diagrama de casos de uso



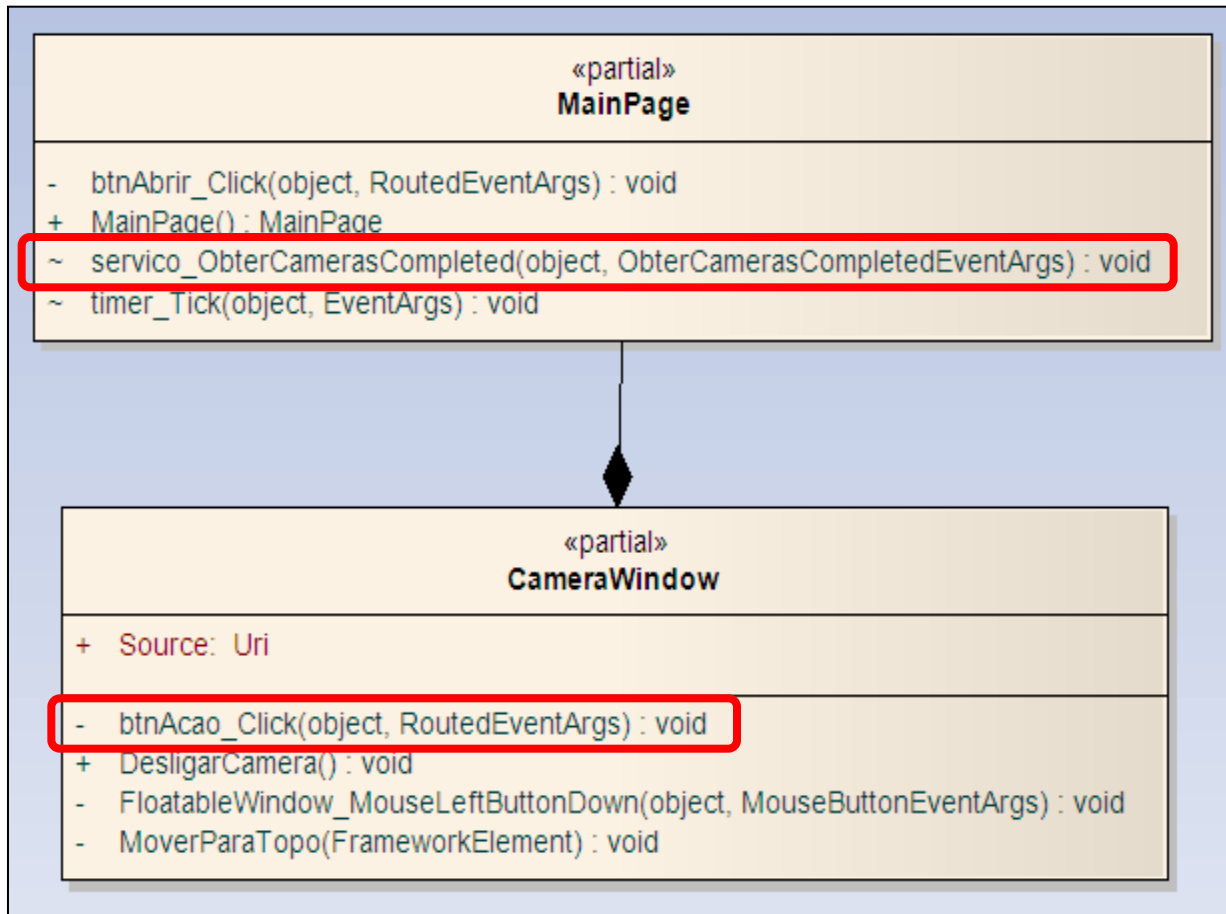
# Diagrama de classes Captura



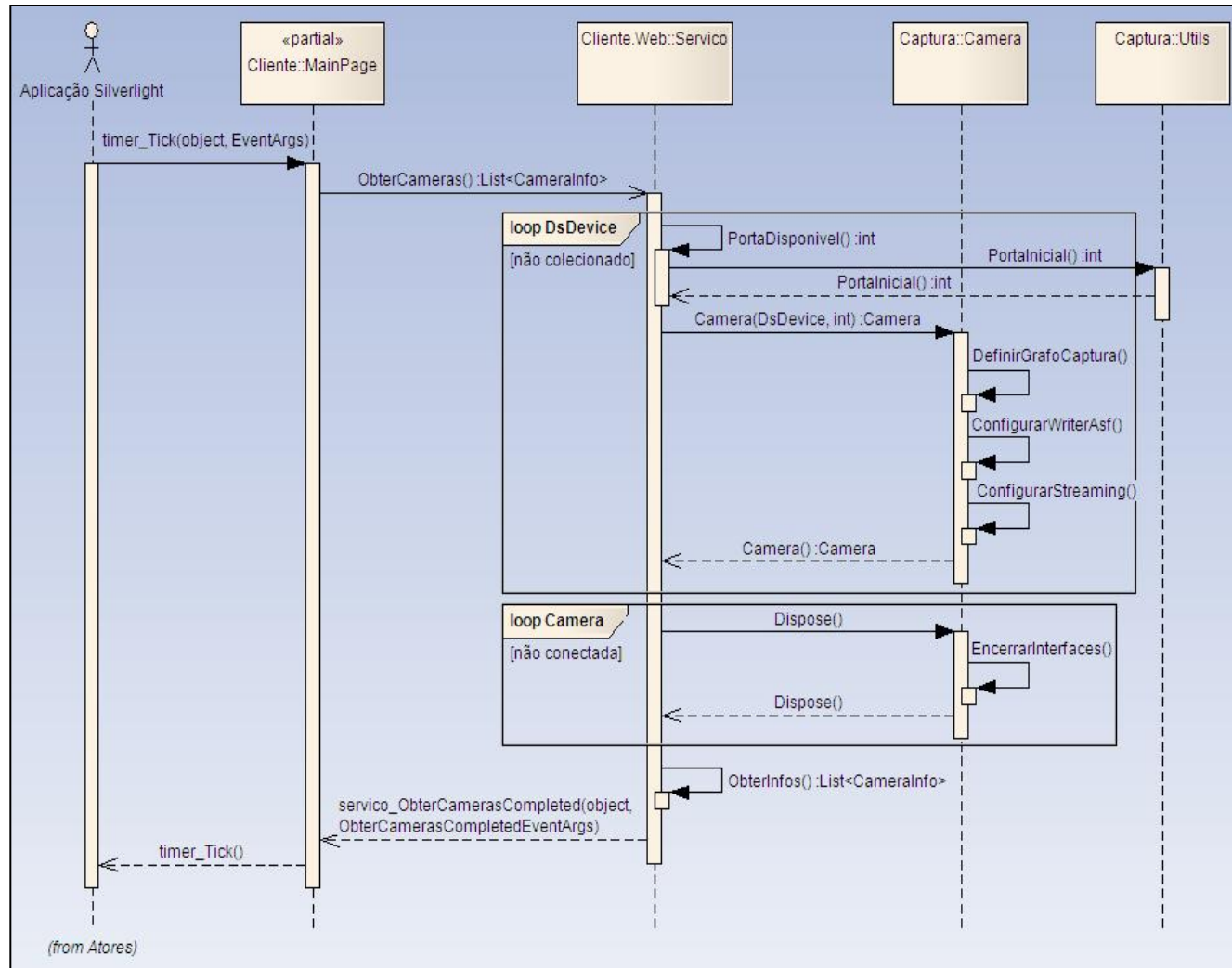
# Diagrama de classes Cliente.Web



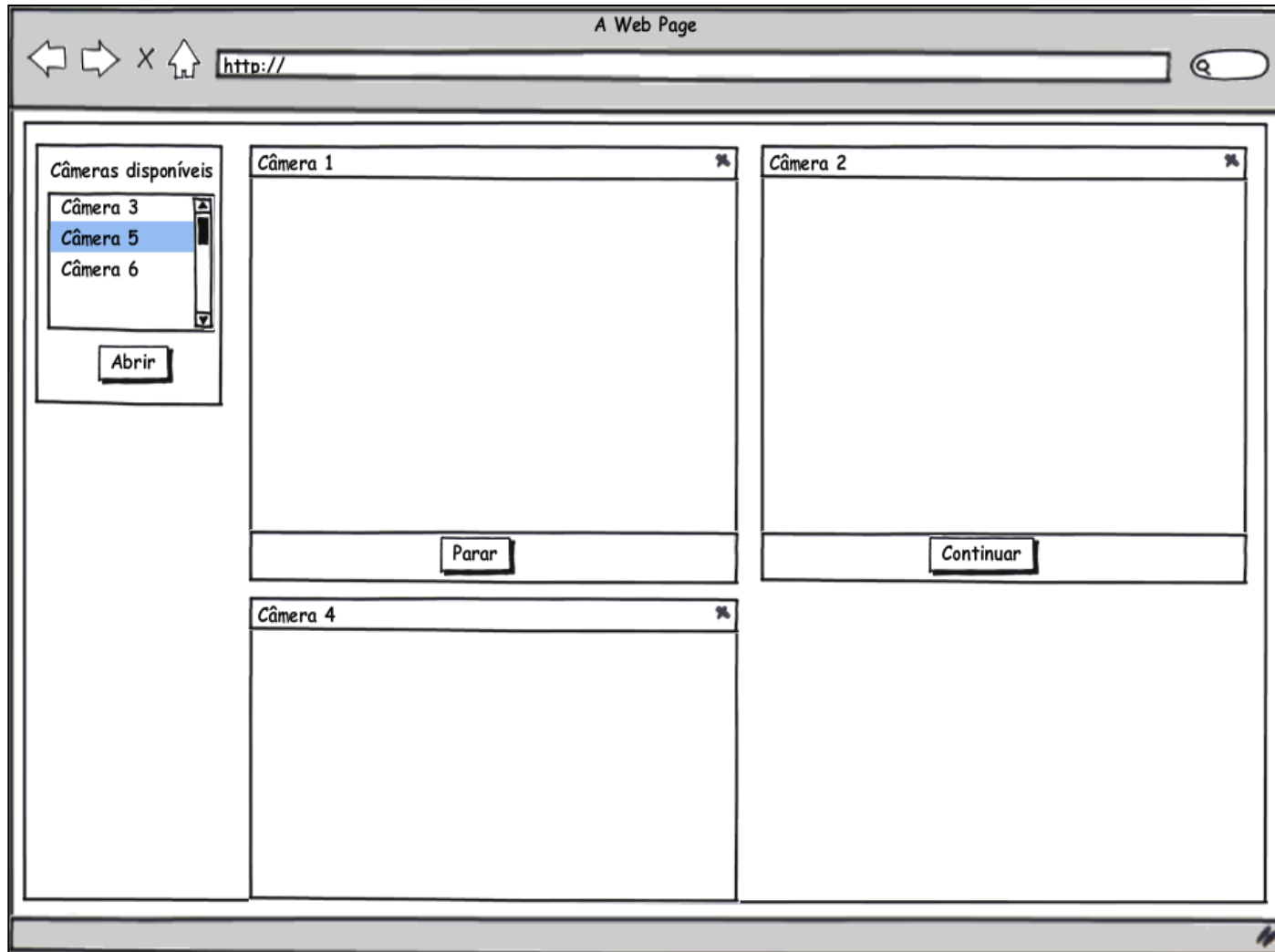
# Diagrama de classes Cliente



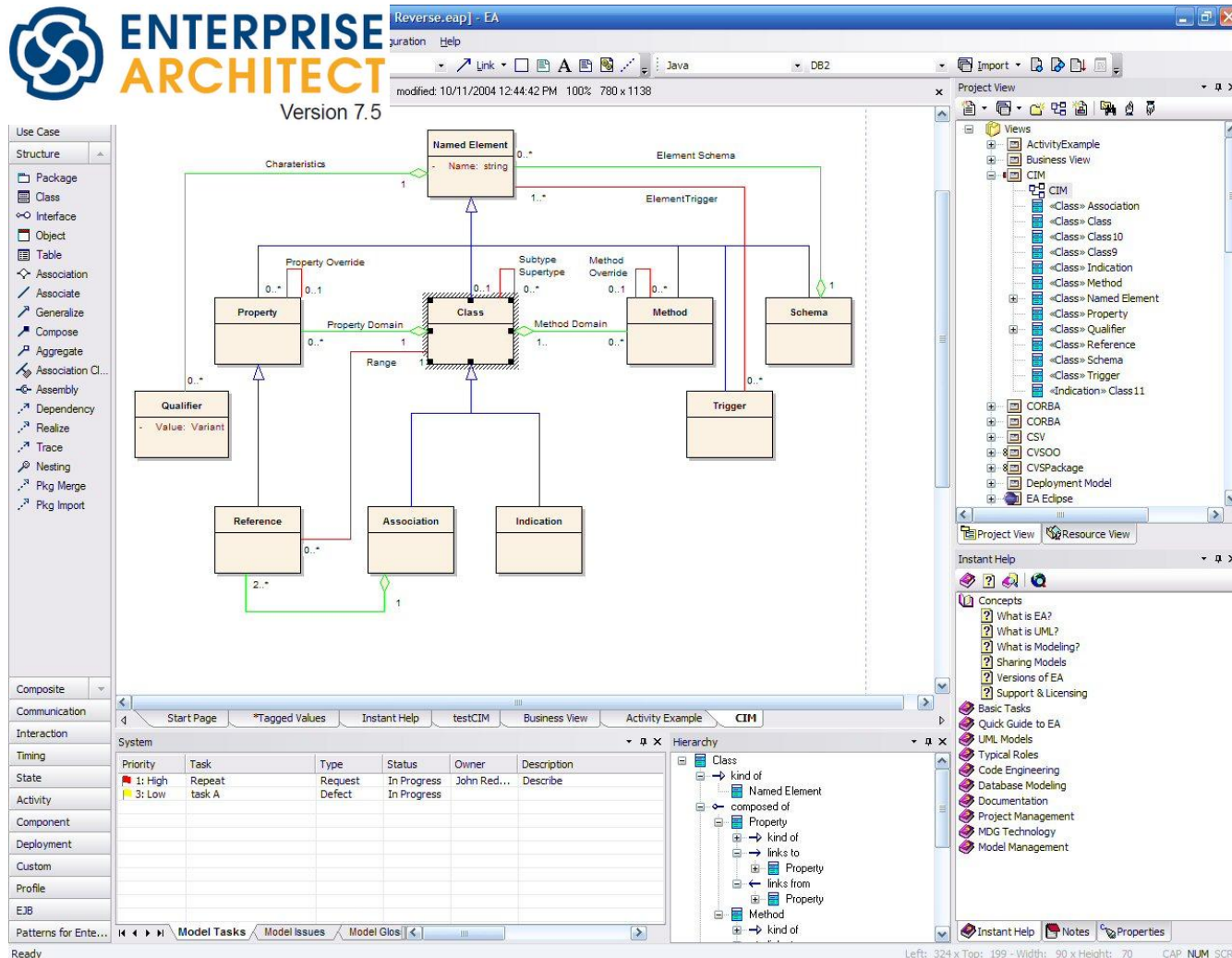
# Diagrama de sequência



# Leiaute da interface



# Tecnologias e ferramentas utilizadas



# Tecnologias e ferramentas utilizadas



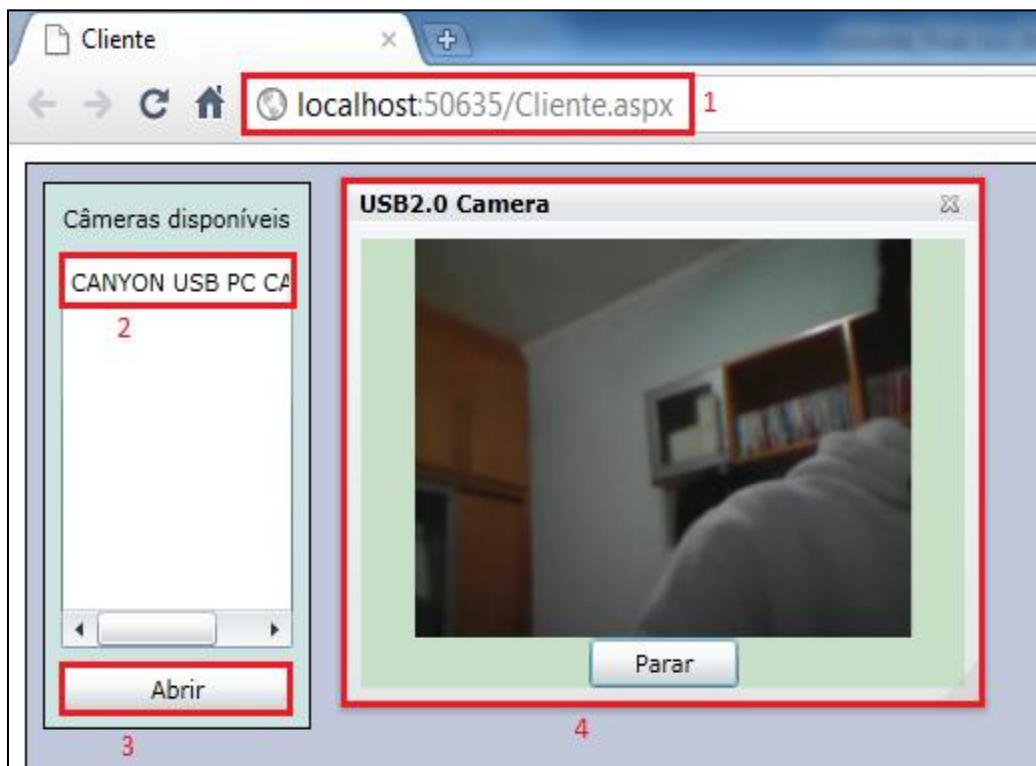
The screenshot displays the Microsoft Visual Studio 2008 environment. The central design area shows a web page layout with a 'Login to Site' form, a 'Welcome to My Site' header, and content sections like 'What's New' and 'Cool Links'. The Properties window on the right shows the 'whatsnew' DIV properties, and the CSS Properties window shows the applied styles. The Source window at the bottom shows the HTML code for the selected element.



# Tecnologias e ferramentas utilizadas

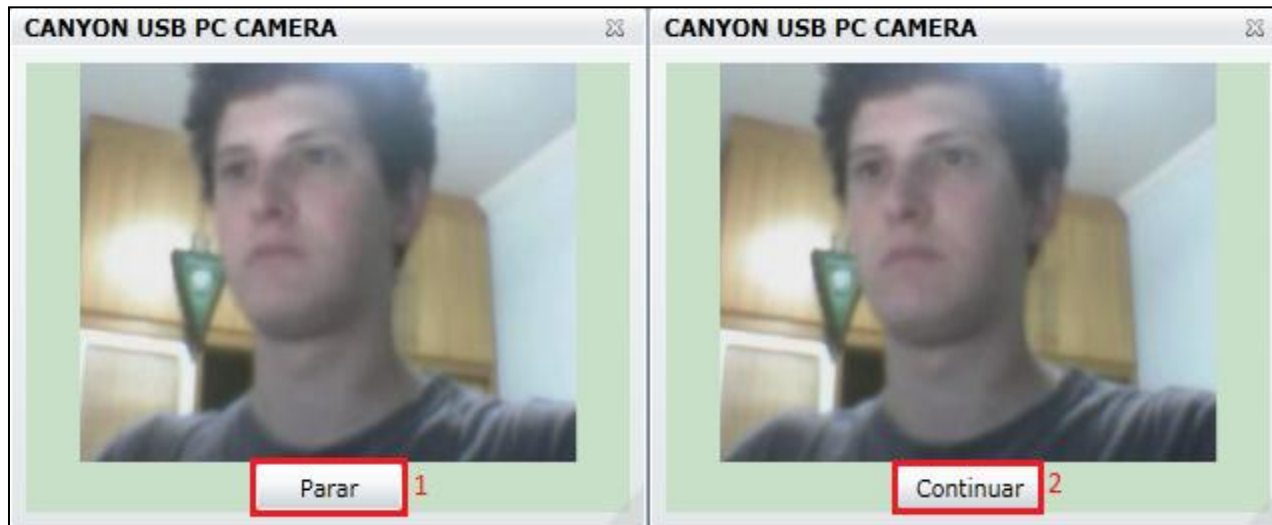


# Operacionalidade da implementação



Operacionalidade do caso de uso UC01

# Operacionalidade da implementação



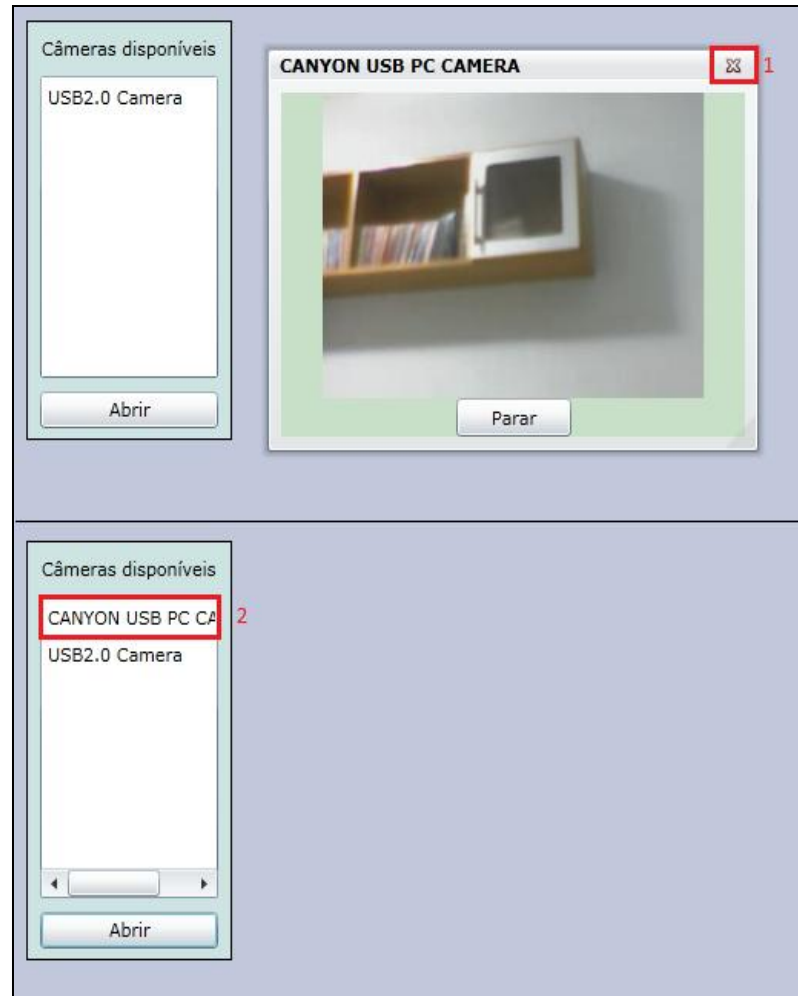
Operacionalidade do caso de uso UC02

# Operacionalidade da implementação



Operacionalidade do caso de uso UC03

# Operacionalidade da implementação



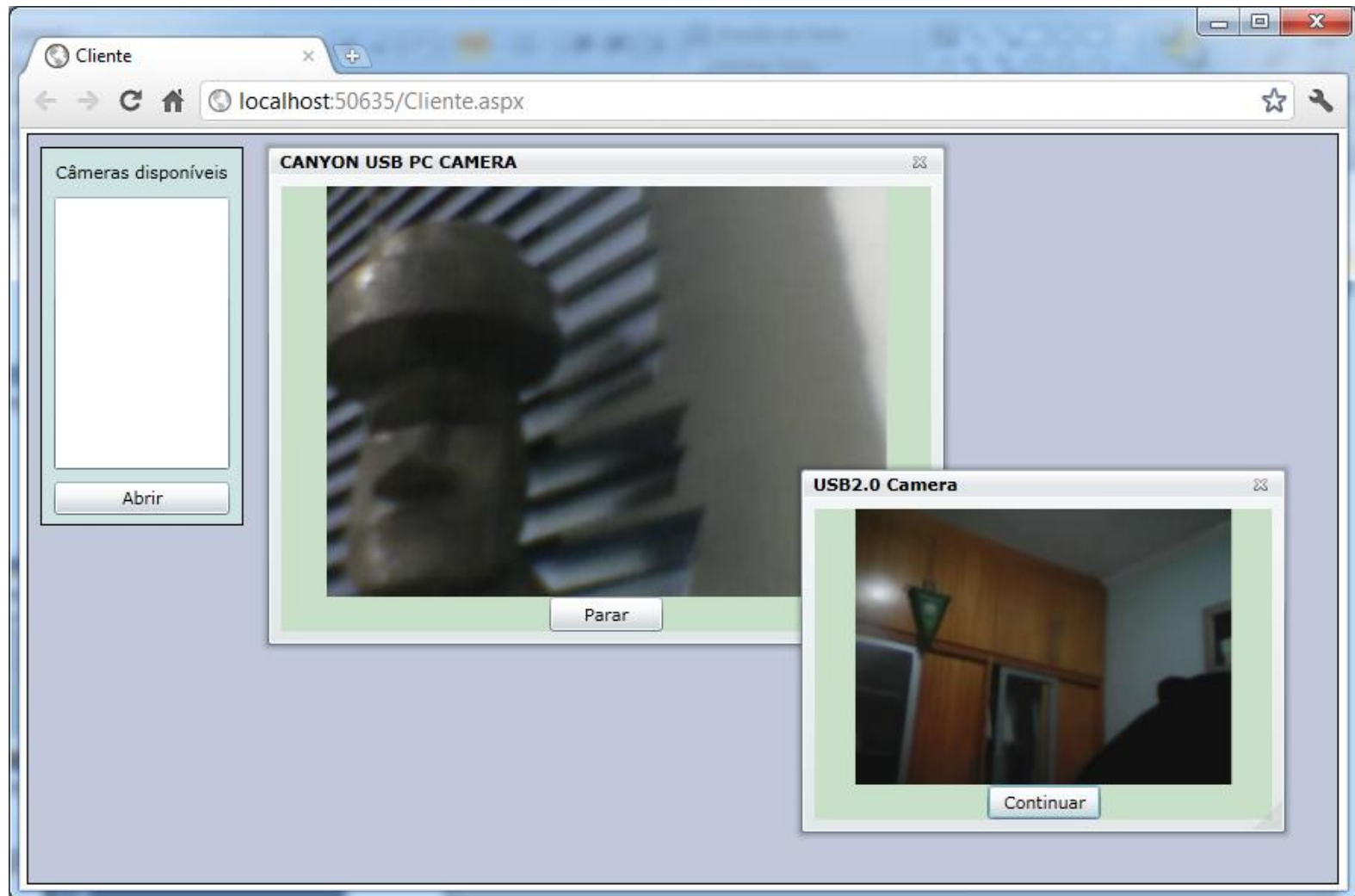
Operacionalidade do caso de uso UC04

# Operacionalidade da implementação



Operacionalidade do caso de uso UC05

# Resultados e discussão



# Resultados e discussão



- ✓ RF01: A aplicação cliente deve disponibilizar um painel que ocupe o maior espaço possível da janela do navegador onde o usuário poderá interagir.



# Resultados e discussão



✓ RF02: A aplicação cliente deve disponibilizar um quadro para cada *streaming* de câmera que estiver recebendo.

# Resultados e discussão



- ✓ RF03: A aplicação cliente deve possuir um menu onde podem ser escolhidas as câmeras escondidas que se deseja restaurar.

# Resultados e discussão



- ✓ RF04: A aplicação cliente deve permitir redimensionamento e movimentação livres (limitado pelo tamanho do painel) de cada quadro de câmera.

# Resultados e discussão



✓ RF05: A aplicação cliente deve permitir fechar cada quadro de câmera.

# Resultados e discussão



- ✓ RF06: A aplicação cliente deve permitir parar e continuar a exibição das imagens de cada janela de câmera.

# Resultados e discussão



✓ RF07: A aplicação servidora deve ler a entrada de cada câmera conectada ao computador.

# Resultados e discussão



✓ RF08: A aplicação servidora deve processar os dados e criar o *streaming* de vídeo de cada câmera.

# Resultados e discussão



✓ RF09: A aplicação servidora deve disponibilizar os *streamings* para leitura na internet.



# Resultados e discussão



Características técnicas dos equipamentos utilizados:

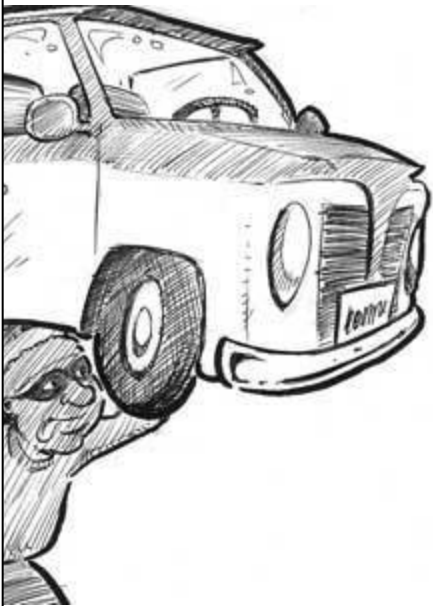
- computador *notebook* modelo Dell Vostro 1310;
- sistema operacional Microsoft Windows 7 Professional 64-bit Service Pack 1;
- processador Intel Core 2 Duo T5670 1.8 Ghz;
- memória RAM 4.0 gigabytes DDR 2 332 Mhz;
- placa de vídeo NVIDIA GeForce 8400M GS;
- resolução de vídeo 1280 x 800 *pixels* com taxa de atualização de 60 Hz e profundidade de cores de 32 *bits*;
- disco rígido de 80 Gb;
- câmera USB com resolução de vídeo 320 x 240 *pixels* e captura de 30 quadros por segundo;
- câmera USB com resolução de vídeo 640 x 480 *pixels* e captura de 30 quadros por segundo.

# Resultados e discussão



Funcionalidade	Este protótipo	Trabalho de Carlassara	WebCam Monitor 5.24 de DeskShare Incorporated
Armazenamento das imagens		✓	✓
Compressão de vídeo	✓	✓	✓
Detecção automática de novas câmeras	✓		
Detecção de movimentos			✓
Movimentação remota das câmeras			✓
Transmissão através da internet	✓	✓	✓
Transmissão de áudio			✓
Transmissão em tempo real	✓		✓
Visualização de múltiplas câmeras simultaneamente	✓		✓
Visualização em dispositivo móvel		✓	
Visualização no navegador	✓		✓

# Conclusões

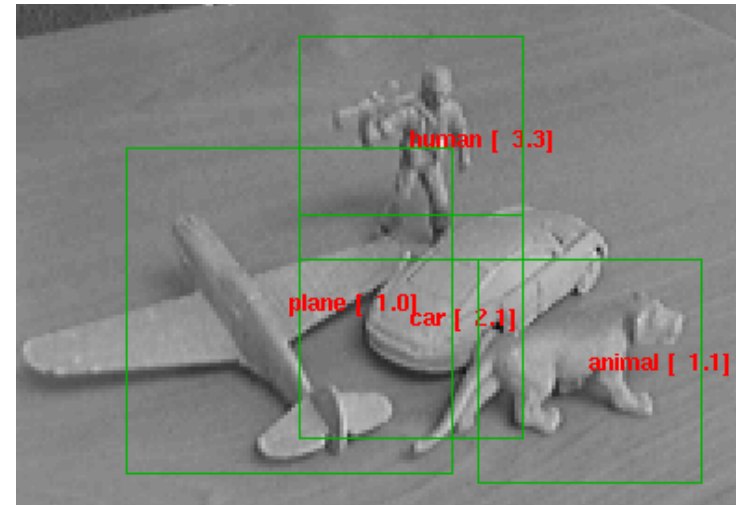
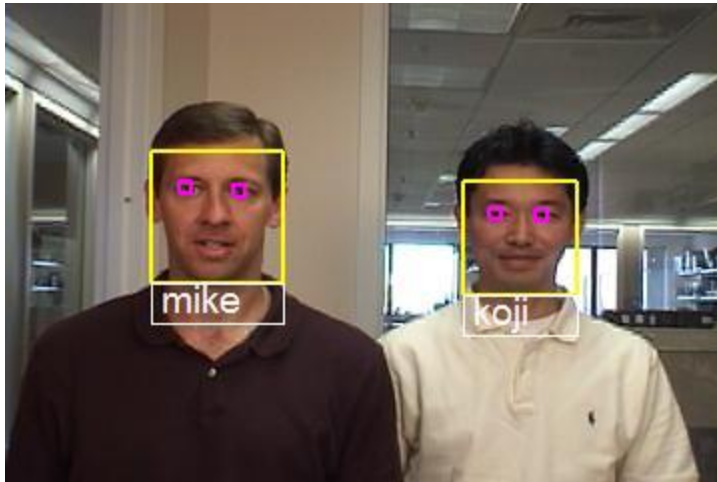


# Conclusões

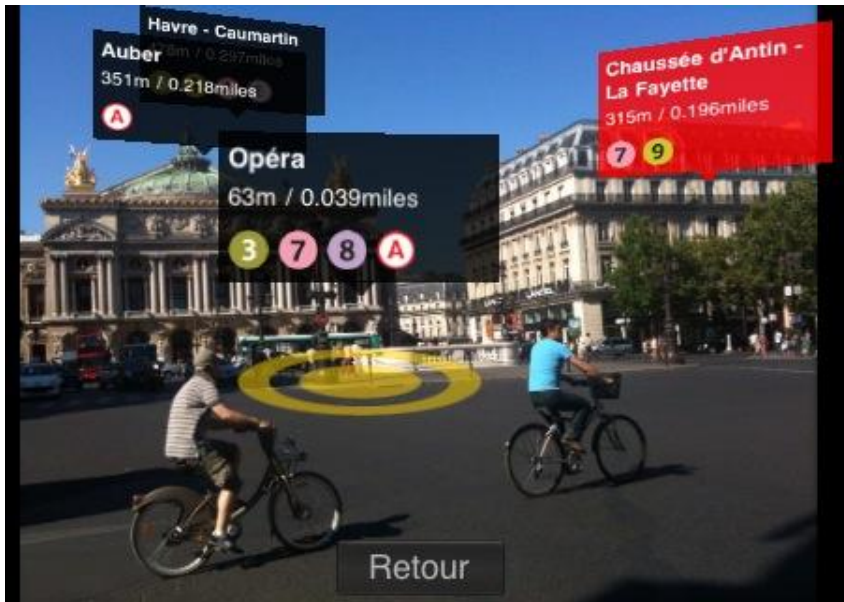


- Utilização de tecnologias RIA, compressão de vídeo e *streaming*
- Dificuldade em integrar os componentes do DirectShow.NET
- Documentação escassa a respeito das tecnologias de compressão e *streaming*
- Orientação a objetos e estratégia de divisão das tarefas do sistema em módulos

# Extensões



# Extensões



- Interface utilizando a tecnologia HTML 5
- Controlar remotamente o ângulo de visão de cada câmera
- Capturar e transmitir áudio
- Controlar o *zoom* da imagem produzida em cada câmera.
- Gravar as imagens obtidas de cada câmera.



Obrigado!