



AAMO Framework

Manual do desenvolvedor

Versão do manual: 1.0
Versão do AAMO: 1.0
Autor: Cleuton Sampaio



<http://www.thecodebakers.org>



CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS LICENSE DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE INFORMATION PROVIDED, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM ITS USE.

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED. BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

1. **"Adaptation"** means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
2. **"Collection"** means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined below) for the purposes of this License.
3. **"Creative Commons Compatible License"** means a license that is listed at <http://creativecommons.org/compatiblelicenses> that has been approved by Creative Commons as being essentially equivalent to this License, including, at a minimum, because that license: (i) contains terms that have the same purpose, meaning and effect as the License Elements of this License; and, (ii) explicitly permits the relicensing of adaptations of works made available under that license under this License or a Creative Commons jurisdiction license with the same License Elements as this License.
4. **"Distribute"** means to make available to the public the original and copies of the Work or Adaptation, as appropriate, through sale or other transfer of ownership.
5. **"License Elements"** means the following high-level license attributes as selected by Licensor and indicated in the title of this License: Attribution, ShareAlike.
6. **"Licensor"** means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
7. **"Original Author"** means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.
8. **"Work"** means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other



writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.

9. **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
10. **"Publicly Perform"** means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
11. **"Reproduce"** means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.

2. Fair Dealing Rights. Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

1. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections;
2. to create and Reproduce Adaptations provided that any such Adaptation, including any translation in any medium, takes reasonable steps to clearly label, demarcate or otherwise identify that changes were made to the original Work. For example, a translation could be marked "The original work was translated from English to Spanish," or a modification could indicate "The original work has been modified.";
3. to Distribute and Publicly Perform the Work including as incorporated in Collections; and,
4. to Distribute and Publicly Perform Adaptations.
5. For the avoidance of doubt:
 - a. **Non-waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
 - b. **Waivable Compulsory License Schemes.** In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor waives the exclusive right to collect such royalties for any exercise by You of the rights granted under this License; and,
 - c. **Voluntary License Schemes.** The Licensor waives the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License.

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. Subject to Section 8(f), all rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:



1. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested. If You create an Adaptation, upon notice from any Licensor You must, to the extent practicable, remove from the Adaptation any credit as required by Section 4(c), as requested.
2. You may Distribute or Publicly Perform an Adaptation only under the terms of: (i) this License; (ii) a later version of this License with the same License Elements as this License; (iii) a Creative Commons jurisdiction license (either this or a later license version) that contains the same License Elements as this License (e.g., Attribution-ShareAlike 3.0 US); (iv) a Creative Commons Compatible License. If you license the Adaptation under one of the licenses mentioned in (iv), you must comply with the terms of that license. If you license the Adaptation under the terms of any of the licenses mentioned in (i), (ii) or (iii) (the "Applicable License"), you must comply with the terms of the Applicable License generally and the following provisions: (I) You must include a copy of, or the URI for, the Applicable License with every copy of each Adaptation You Distribute or Publicly Perform; (II) You may not offer or impose any terms on the Adaptation that restrict the terms of the Applicable License or the ability of the recipient of the Adaptation to exercise the rights granted to that recipient under the terms of the Applicable License; (III) You must keep intact all notices that refer to the Applicable License and to the disclaimer of warranties with every copy of the Work as included in the Adaptation You Distribute or Publicly Perform; (IV) when You Distribute or Publicly Perform the Adaptation, You may not impose any effective technological measures on the Adaptation that restrict the ability of a recipient of the Adaptation from You to exercise the rights granted to that recipient under the terms of the Applicable License. This Section 4(b) applies to the Adaptation as incorporated in a Collection, but this does not require the Collection apart from the Adaptation itself to be made subject to the terms of the Applicable License.
3. If You Distribute, or Publicly Perform the Work or any Adaptations or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (e.g., a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and (iv) , consistent with Section 3(b), in the case of an Adaptation, a credit identifying the use of the Work in the Adaptation (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Adaptation or Collection, at a minimum such credit will appear, if a credit for all contributing authors of the Adaptation or Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.



4. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Adaptations or Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation. Licensor agrees that in those jurisdictions (e.g. Japan), in which any exercise of the right granted in Section 3(b) of this License (the right to make Adaptations) would be deemed to be a distortion, mutilation, modification or other derogatory action prejudicial to the Original Author's honor and reputation, the Licensor will waive or not assert, as appropriate, this Section, to the fullest extent permitted by the applicable national law, to enable You to reasonably exercise Your right under Section 3(b) of this License (right to make Adaptations) but not otherwise.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

1. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Adaptations or Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
2. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

1. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
2. Each time You Distribute or Publicly Perform an Adaptation, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.
3. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
4. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
5. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
6. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic



Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.

Creative Commons Notice

Creative Commons is not a party to this License, and makes no warranty whatsoever in connection with the Work. Creative Commons will not be liable to You or any party on any legal theory for any damages whatsoever, including without limitation any general, special, incidental or consequential damages arising in connection to this license. Notwithstanding the foregoing two (2) sentences, if Creative Commons has expressly identified itself as the Licensor hereunder, it shall have all rights and obligations of Licensor.

Except for the limited purpose of indicating to the public that the Work is licensed under the CCPL, Creative Commons does not authorize the use by either party of the trademark "Creative Commons" or any related trademark or logo of Creative Commons without the prior written consent of Creative Commons. Any permitted use will be in compliance with Creative Commons' then-current trademark usage guidelines, as may be published on its website or otherwise made available upon request from time to time. For the avoidance of doubt, this trademark restriction does not form part of the License. Creative Commons may be contacted at <http://creativecommons.org/>.



- [License](#)
- [Creative Commons Notice](#)
- [Introdução](#)
 - [Estrutura básica de uma aplicação AAMO](#)
 - [O que é necessário para desenvolver apps AAMO](#)
 - [Sobre a linguagem Lua](#)
- [Como criar um novo projeto AAMO](#)
 - [Criando um projeto Android](#)
 - [Criando um projeto iOS](#)
- [Internacionalização e Localização](#)
- [Arquivo de Tela](#)
 - [UI](#)
 - [MENU](#)
 - [Eventos](#)
 - [Funções](#)
 - [LABEL](#)
 - [Eventos](#)
 - [Funções](#)
 - [TEXTBOX](#)
 - [Eventos](#)
 - [Funções](#)
 - [BUTTON](#)
 - [Eventos](#)
 - [Funções](#)
 - [CHECKBOX](#)
 - [Eventos](#)
 - [Funções](#)
 - [LISTBOX](#)
 - [Eventos](#)
 - [Funções](#)
 - [WEBBOX](#)
 - [Eventos](#)
 - [Funções](#)
 - [IMAGEBOX](#)
 - [Eventos](#)
 - [Funções](#)
- [API de programação](#)
 - [addListBoxOption](#)
 - [clearListBox](#)
 - [close](#)
 - [closeDatabase](#)
 - [eof](#)
 - [execSQL](#)
 - [exitScreen](#)
 - [getCheckBox](#)
 - [getCurrentScreenId](#)
 - [getError](#)
 - [getGlobalParameter](#)
 - [getLabelText](#)
 - [getLocalizedText](#)
 - [getTextField](#)



[loadScreen](#)
[log](#)
[navigateTo](#)
[next](#)
[openDatabase](#)
[query](#)
[setCheckBox](#)
[setGlobalParameter](#)
[setLabelText](#)
[setPicture](#)
[setTextField](#)
[showMenu](#)
[showMessage](#)
[showScreen](#)
[Acesso a bancos de dados](#)



Introdução

“Aamo” quer dizer “Lua” no idioma dos índios Xavantes.

O AAMO é um framework multiplataforma para criação de aplicações para dispositivos móveis, como os baseados em Android (Google) ou iOS (Apple).

Com ele, você cria sua aplicação de maneira simples e rápida, utilizando linguagem não verbosa, e pode executá-la imediatamente em várias plataformas móveis.

Uma aplicação AAMO é composta por arquivos XML, que descrevem as telas e seus controles (interface do usuário) e, opcionalmente, arquivos contendo código-fonte em Lua. Lua é uma linguagem moderna, criada para ser executada em modo “embarcado” (dentro de outros softwares), pequena, rápida e completa. Além disto, é utilizada por vários frameworks e produtos pelo mundo afora. E, se não bastar, é Nacional (PUC-RJ) e Open Source.

Estrutura básica de uma aplicação AAMO

A aplicação mais simples pode ser composta apenas por um único arquivo XML. Vamos criar um “Hello World” utilizando o AAMO. Para começar, crie um arquivo texto e salve-o como: “ul.xml”. Depois, digite o seguinte conteúdo:

```
<?xml version="1.0" encoding="UTF-8"?>
<ui>
  <version>0.3</version>
  <uuid>1</uuid>
  <title>Teste AAMO</title>
  <element>
    <id>1</id>
    <type>2</type>
    <percentTop>0</percentTop>
    <percentLeft>0</percentLeft>
    <percentWidth>20</percentWidth>
    <percentHeight>10</percentHeight>
    <text>Nome</text>
  </element>
  <element>
    <id>2</id>
    <type>1</type>
    <percentTop>0</percentTop>
    <percentLeft>21</percentLeft>
    <percentWidth>35</percentWidth>
    <percentHeight>10</percentHeight>
  </element>
  <element>
    <id>3</id>
    <type>3</type>
    <percentTop>20</percentTop>
    <percentLeft>40</percentLeft>
    <percentWidth>20</percentWidth>
    <percentHeight>10</percentHeight>
    <text>Ok</text>
    <onClickScript>lua::aamo.showMessage("Olá " .. aamo.getTextField(2))</onClickScript>
  </element>
</ui>
```



Ao ser executada, esta aplicação exibe a seguinte tela:



Você pode rodá-la em qualquer dispositivo para o qual o AAMO esteja disponível. Atualmente, funciona apenas em Android e iOS (iPad, iPhone e iPod touch).

(veja no capítulo sobre instalação como rodar esta aplicação no dispositivo de sua preferência)

Vamos entender a estrutura. Para começar, é preciso conhecer o “idioma” do AAMO, ou seja, quais são os macro-objetos com os quais ele trabalha.

Aplicação

Uma aplicação AAMO é um conjunto de arquivos texto, todos contidos na pasta “app” (no Android: “/assets/app”, no iOS é o grupo “app”).

Uma aplicação pode ser composta por uma ou mais telas (arquivos em XML), zero ou mais arquivos de tratamento de evento (arquivos “*.lua”), zero ou mais arquivos de esquema de banco de dados (em XML) e zero ou mais arquivos de tradução (“properties”).

Uma aplicação AAMO é totalmente portátil entre as plataformas, bastando copiar o conteúdo da pasta “app”. Não é necessário alterar a aplicação para que funcione em Android ou iOS.

Tela



Para o AAMO tudo o que o usuário vê no “display” do dispositivo, em determinado momento, é uma “Tela” (ou “Screen”). Chega de nomes complicados para coisas simples (“View”, “Activity” etc).

Uma aplicação pode ter uma ou mais telas. Para que seja executada pelo AAMO deve possuir pelo menos uma tela.

A primeira tela tem o identificador 1 (<uuid>1</uuid>) e deve se chamar “ui.xml” obrigatoriamente. Além disto, ela é carregada automaticamente pelo AAMO.

As demais telas podem ser numeradas a partir de 2: “ui_2.xml”, “ui_3.xml” etc. O número da tela deve constar do tag “<uuid>” e do nome do arquivo, por exemplo, o arquivo “ui_2.xml” deve ter um tag “<uuid>2</uuid>”.

Controle

Uma tela pode ter um ou mais controles. Um controle é um elemento visual com o qual o usuário interage. Atualmente existem 7 tipos de controles:

- TEXTBOX (tipo 1): campo para digitação de texto;
- LABEL (tipo 2): rótulo;
- BUTTON (tipo 3): botão tocável;
- CHECKBOX (tipo 4): caixa de opção binária;
- LISTBOX (tipo 5): lista de elementos selecionáveis;
- WEBBOX (tipo 6): mini-navegador web;
- IMAGEBOX (tipo 7): exibidor de imagens;

Um controle é identificado por seu “id”, que é um número sequencial (iniciando-se de 1). Não pode haver dois controles com mesmo “id” em uma mesma tela.

Eventos e tratamento

O AAMO é baseado em eventos. O código da sua aplicação é executado em função dos eventos. Para tratar os eventos existem duas maneiras: 1) você escreve o tratamento dentro do próprio arquivo XML; ou 2) você escreve um arquivo de tratamento (com extensão “lua”).

Analisando a estrutura da aplicação exemplo

A pequena aplicação que mostramos nos permitirá analisar e entender muita coisa da estrutura do AAMO. Para começar, vamos ver os elementos de definição da tela:

```
<ui>
  <version>0.3</version>
  <uuid>1</uuid>
  <title>Teste AAMO</title>
```

Estamos definindo uma tela inicial, que é carregada automaticamente no início da aplicação. O tag “<version>” indica qual é a versão mínima do AAMO para executar esta aplicação, e o tag “<uuid>” indica o número da tela, que deve constar do nome do arquivo. Para a primeira tela, isto não vale, pois o nome tem que ser obrigatoriamente: “ui.xml”. Para as demais, o número da tela deve vir após um caracter “_”, por exemplo: “ui_2.xml”.

Existem mais elementos que podemos definir para a tela, como: tratadores de eventos, menu e cor de fundo, mas vamos deixar para depois.

Os controles são declarados com o tag “<element>”, através do qual indicamos o “id” e o tipo de controle, além de configurar seu posicionamento e tamanho. Por exemplo:



```
<element>
  <id>1</id>
  <type>2</type>
  <percentTop>0</percentTop>
  <percentLeft>0</percentLeft>
  <percentWidth>20</percentWidth>
  <percentHeight>10</percentHeight>
  <text>Nome</text>
</element>
```

Aqui declaramos um controle do tipo “LABEL” (tipo 2), e indicamos seu conteúdo (“<text>Nome</text>”). Porém, as principais propriedades são:

- “percentTop”: posição em relação ao topo da tela;
- “percentLeft”: posição em relação à esquerda da tela;
- “percentWidth”: largura com relação à largura da tela;
- “percentHeight”: altura com relação à altura da tela;

Tudo é definido através de percentuais com relação ao tamanho da tela. Isto garante que uma aplicação AAMO funcione de maneira proporcional em qualquer tipo de dispositivo, desde um smartphone até um tablet. Os tags “<percentTop>” e “<percentLeft>” indicam as coordenadas do canto superior esquerdo do controle, lembrando que, em dispositivos móveis, o valor da abscissa é zero na parte superior e cresce no sentido para baixo. Assim, o canto superior esquerdo é posicionado com base no percentual da altura da tela (percentTop) e no percentual da largura da tela (percentLeft).

Os outros dois tags indicam a largura (percentWidth) e altura (percentHeight) do controle, tudo com base em percentuais da largura e altura da tela do dispositivo.

Alguns tipos de controles, como LABEL e TEXTBOX, possuem propriedades extras, como “<text>”, além de tratadores de eventos. Na tela temos um exemplo disto:

```
<element>
  <id>3</id>
  <type>3</type>
  <percentTop>20</percentTop>
  <percentLeft>40</percentLeft>
  <percentWidth>20</percentWidth>
  <percentHeight>10</percentHeight>
  <text>Ok</text>
  <onClickScript>lua::aamo.showMessage("Olá " .. aamo.getTextField(2))</onClickScript>
</element>
```

Este é um controle do tipo 3, um BUTTON. Além da propriedade “text”, ele permite também adicionarmos um tratador de evento: “onClickScript”. Quando o usuário tocar no botão, o AAMO vai invocar o tratamento especificado na propriedade “onClickScript”. Neste caso, é um comando imediato:

```
aamo.showMessage("Olá " .. aamo.getTextField(2))
```

O tratamento de eventos pode consistir de comandos imediatos, que são armazenados no próprio arquivo XML da tela, ou de rotinas externas, escritas em Lua. Por exemplo, poderíamos ter modificado o programa para usar uma rotina externa. No arquivo “ui.xml” especificaríamos:

```
<element>
  <id>3</id>
```



```
<type>3</type>
<percentTop>20</percentTop>
<percentLeft>40</percentLeft>
<percentWidth>20</percentWidth>
<percentHeight>10</percentHeight>
<text>Ok</text>
<onClickScript>clcou</onClickScript>
</element>
```

E criaríamos uma rotina externa chamada “clcou.lua”, com o seguinte conteúdo:

```
aamo.showMessage("Olá " .. aamo.getTextField(2))
```

O que diferencia um do outro é a presença do prefixo “lua::” no início do conteúdo do tag.

O que é necessário para desenvolver apps AAMO

Você precisa desenvolver a aplicação, embuti-la dentro de um “projeto-berço”, e distribuir conforme as regras específicas de cada plataforma (Android ou iOS).

O AAMO utiliza o conceito de “projeto-berço”, que são projetos nativos de cada plataforma. Você copia o projeto (mostraremos como deve ser feito) e inclui sua aplicação na pasta “app”. Pronto! Não há necessidade de modificar quase nada no projeto-berço. Na verdade, você sequer precisa conhecer as linguagens e ambientes nativos de cada plataforma.

O AAMO é open source e gratuito, logo, você não precisa comprar licença de desenvolvimento e distribuição. A única exigência é que você distribua os arquivos de licença (...license.txt) que vêm junto com cada projeto-berço.

Sobre a linguagem Lua

Neste manual não ensinaremos a linguagem Lua. Se você quiser aprender, existem vários livros e tutoriais na Internet, entre eles:

- <http://www.lua.org/portugues.html>;
- <http://www.lua.org/doc/jai2009.pdf>

Como criar um novo projeto AAMO

O AAMO é absolutamente imparcial quanto à plataforma móvel. Você pode desenvolver utilizando Android e depois executar utilizando iOS, sem problema algum, e sem necessidade de modificar sua aplicação.

Se você utilizar o AAMO Editor, que é a IDE do AAMO, já contará com um projeto-berço embutido, que roda em desktop, logo, não há necessidade de baixar um projeto-berço para desenvolver sua aplicação. O AAMO Editor não é assunto deste manual, logo, se tiver interesse, sugerimos consultar a documentação apropriada nos sites:

- <http://www.ammoframework.org> (versão com erro de DNS);
- <http://www.aamoframework.org>;
- <http://www.thecodebakers.org>;
- <http://code.google.com/p/aamoframework/>;

Criando um projeto Android

Baixe o projeto-berço Android (veja em: <http://code.google.com/p/aamoframework/downloads/list>), abra a IDE “eclipse”, devidamente configurada com o plugin “ADT”, e selecione “import / general / Existing projects into Workspace”, apontando pra o ZIP do AAMO.

Ao fazer isto, você terá uma cópia do projeto-berço, com o nome e a configuração originais do AAMO. Se for só para desenvolver ou fazer um teste rápido, ok. Mas se você quiser distribuir seu projeto, é melhor mudar algumas coisas:

1. Mude o nome do pacote da aplicação:
 - a. Selecione seu projeto com o botão direito, abra o menu “Android Tools” e selecione “Rename Application Package”;
 - b. Informe o novo nome de pacote. Atenção: este é o nome da sua aplicação no Google Play, preste atenção!
2. Altere a versão da sua aplicação no AndroidManifest.xml:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="org.thecodebakers.testeaamo"
    android:versionCode="1"
    android:versionName="1.0" >
```

Nesta versão (0.3 alpha ou 1.0), o AAMO somente trabalha com a versão da API Android 2.2. Depois disto é só copiar TODOS os arquivos da sua aplicação para a pasta “assets/app”.

Criando um projeto iOS



Baixe o projeto-berço iOS (veja em: <http://code.google.com/p/aamoframework/downloads/list>), descompactando-o na pasta de sua preferência. Abra o Xcode e abra o projeto nele. Na janela "Project Navigator" clique duas vezes devagar sobre o nome da pasta raiz do projeto (a primeira, em azul), e troque seu nome.

Se quiser, selecione o Target e troque o nome do pacote, dentro do "Bundle identifier". Aproveite e altere também os campos "Version" e "Build".

Abra o grupo "app" e apague todos os arquivos que estiverem lá, depois copie os arquivos da sua aplicação.



Internacionalização e Localização

Internacionalização (i18n) é o processo de preparar uma aplicação para utilizar múltiplos idiomas e preferências, de acordo com culturas diferentes. Localização (L10n) é o processo de adaptar uma aplicação internacionalizada a uma linguagem específica.

O AAMO dá suporte a ambas as técnicas de maneira simples e transparente. Para começar, podemos criar arquivos de tradução, especificando o idioma e o país. Estes arquivos são como arquivos de propriedades, e tem a sintaxe:

<nome do texto> = <valor do texto naquele idioma>

Por exemplo:

```
teste=Teste Traduzido
locmsg=Mensagem Default
loclista=Praia
first=(PT) Primeiro
murl=http://pt.wikipedia.org/wiki/Brasil
```

Tanto no arquivo XML de tela, como no código Lua podemos nos referir a textos localizados. Por exemplo, podemos criar um LABEL com um texto localizado:

```
<text>l10n: teste</text>
```

Sempre que o AAMO reconhecer o prefixo “**l10n**” (L + 10 + N) em uma propriedade, ele buscará o texto no arquivo de tradução ativo.

As propriedades que aceitam tradução atualmente são:

- text: BUTTON, LABEL, TEXTBOX;
- url: WEBBOX;
- picture: IMAGEBOX;
- option: <menu>;

Dentro do código-fonte Lua, podemos utilizar a função “aamo.getLocalizedText()”:

texto = aamo.getLocalizedText(<nome_do_string>)

O arquivo de tradução tem o nome básico:

```
aamol10n.properties
```

Este arquivo deve ficar sempre junto com os outros arquivos da aplicação AAMO, dentro da pasta “app”.

Podemos criar arquivos por idioma, por exemplo:

- aamol10n_en.properties;



- `aamol10n_pt.properties;`
- `aamol10n_fr.properties;`

É só acrescentar o sufixo “_” com o código do idioma baseado no padrão ISO 639-1 (http://www.w3schools.com/tags/ref_language_codes.asp).

Também podemos “especializar” a tradução, acrescentando o código ISO 3166 do país (http://userpage.chemie.fu-berlin.de/diverse/doc/ISO_3166.html).

Podemos ter os arquivos:

- `aamol10n_en_GB.properties;`
- `aamol10n_en.properties;`
- `aamol10n.properties;`

O AAMO irá sempre tentar achar o arquivo completo (idioma e país). Se não encontrar, vai procurar o que contenha o idioma e, caso não encontre, vai procurar o default (`aamol10n.properties`). Ele só procurará os arquivos se algum texto localizado for especificado na tela, ou se a função “`aamo.getLocalizedText()`” for invocada. Caso contrário, não é necessário criar arquivos de tradução.

Arquivo de Tela

UI

Todo arquivo de tela AAMO deve começar pelo tag “<ui>”. Os elementos deste tag são:

- “menu”: define um menu para a tela;
- “version”: especifica a versão AAMO para a qual a tela foi escrita;
- “uiid”: o identificador da tela, que deve ser compatível com o seu nome de arquivo;
- “title”: atualmente é só documental;
- “onLoadScript”: tratamento de evento de carga da tela;
- “onEndScript”: tratamento de evento de descarga da tela;
- “onLeaveScript”: tratamento de evento de pausa da tela;
- “onBackScript”: tratamento de evento de reativação da tela;
- “backgroundColor”: cor de fundo da tela;

É preciso entender como o AAMO trata as telas. Ele possui uma “pilha” de telas (LIFO). Quando carregamos uma tela, a tela anterior vai para a penúltima posição. Quando uma tela é descarregada, a tela anterior volta a ser a última da pilha e, conseqüentemente, exibida na tela.

Quando uma tela é carregada, ou seja: não existia na pilha, seu tratamento de evento “onLoadScript” é invocado. Ele só será invocado novamente se ela for descarregada da pilha e carregada novamente. Uma tela descarregada deixa de existir na memória.

Quando uma tela é descarregada, seu tratamento de evento “onEndScript” é invocado. Isto ocorre para todas as telas EXCETO para a tela 1. A tela 1 NUNCA terá seu evento “onEndScript” invocado.

O evento “onLeaveScript” é executado sempre que uma tela invoca outra. Neste caso, a tela original vai para a penúltima posição da pilha e, portanto, será considerada “em pausa”. O seu tratamento “onLeaveScript” será invocado.

O evento “onBackScript” ocorre sempre que uma tela retorna ao topo da pilha, exceto se estiver sendo carregada neste momento. Se a tela “A” invocou a tela “B”, o evento “onLeaveScript” ocorrerá na tela “A”. Ao remover a tela “B” e voltar para a tela “A”, seu evento “onBackScript” será invocado.

Note que os eventos são totalmente opcionais.

A propriedade “backgroundColor” permite especificar a cor de fundo da tela, em formato HTML hexadecimal (http://www.w3schools.com/html/html_colornames.asp). Por exemplo:

- #000000: preto;
- #FFFFFF: branco;
- #FF0000: vermelho;
- #00FF00: verde;
- #0000FF: azul;
- #D3D3D3: cinza claro;



É um formato de cores RGB (Red, Green, Blue), no qual especificamos a quantidade de vermelho, verde e azul em cada cor. Cada dois caracteres especificam a quantidade. Veja no site abaixo quais são as combinações:

http://www.w3schools.com/html/html_colornames.asp

MENU

O Menu é um tipo de controle. Só pode haver um menu por tela. Nós o especificamos dentro do tag “<ui>”. Por exemplo:

```
<ui>
  <menu>
    <option>l10n::first</option>
    <option>Opção Um</option>
    <option>Opção Dois</option>
    <onMenuSelected>selecionado</onMenuSelected>
  </menu>
...
</ui>
```

Cada tag “<option>” estabelece um texto que aparecerá no menu. O tratamento de evento “onMenuSelected” será invocado sempre que o usuário tocar em um item de menu. Neste exemplo, o arquivo “selecionado.lua” será invocado (poderia ser imediato também: lua::).

Note que os itens de menu podem ser internacionalizados.

Só pode haver um menu por tela, e ele tem que ser definido estaticamente no arquivo de tela.

Para exibir o menu, o usuário poderá invocar o comando “aamo.showMenu()”. Se estiver utilizando um dispositivo Android, o próprio botão “menu” acionará o menu. Porém, para compatibilidade, recomendamos que seja criado um botão em todas as telas que usam menu:

```
<element>
  <type>3</type>
  <id>NNN</id>
  <percentTop>YYYYYY</percentTop>
  <percentLeft>XXXXXX</percentLeft>
  <percentHeight>20</percentHeight>
  <percentWidth>30</percentWidth>
  <text>l10n::menuLabel</text>
  <onClickScript>lua::aamo.showMenu()</onClickScript>
</element>
```

Eventos

Quando o usuário clica em um item de menu, o tratamento “onMenuSelected” é invocado. O AAMO cria dois parâmetros globais (veja a definição de parâmetros globais adiante):



- “aamo::selectedMenuIndex”: contém o índice selecionado (começando de zero);
- “aamo::selectedMenuText”: contém o texto selecionado (já traduzido, se for o caso);

Funções

- **aamo.showMenu()** : ao ser invocado exibe o menu configurado para a tela.

LABEL

Um label é um controle “read only”, ou seja, apenas para exibir informações para o usuário. Seu tipo é 2 e pode ser definido desta forma:

```
<element>
  <id>2</id>
  <type>2</type>
  <percentTop>10</percentTop>
  <percentLeft>0</percentLeft>
  <percentWidth>30</percentWidth>
  <percentHeight>10</percentHeight>
  <text>l10n::teste</text>
</element>
```

Note que este LABEL possui um texto localizado, fazendo com que o AAMO procure a tradução da chave “teste” no arquivo de tradução configurado.

As propriedades “percentXXX” foram explicadas na introdução deste manual.

Eventos

O LABEL não responde a eventos.

Funções

- texto = aamo.**getLabelText**(<id do LABEL>): retorna o conteúdo do LABEL;
- aamo.**setLabelText**(<id do LABEL>, “<texto do label>”): altera o conteúdo do LABEL;

TEXTBOX

Uma caixa para digitação de texto. Pode ser definido como:



```
<element>
  <id>1</id>
  <type>1</type>
  <percentTop>10</percentTop>
  <percentLeft>40</percentLeft>
  <percentWidth>40</percentWidth>
  <percentHeight>10</percentHeight>
</element>
```

A TEXTBOX é o controle de tipo 1. Podemos definir seu conteúdo com o tag “text”, que pode ser localizado, ou através de funções Lua.

Eventos

A TEXTBOX não responde a eventos.

Funções

- **texto = aamo.getTextField(<id da TEXTBOX>):** obtém o conteúdo digitado em uma TEXTBOX;
- **aamo.setTextField(<id da TEXTBOX>, “<Texto>”):** altera o conteúdo de uma TEXTBOX;

BUTTON

É um botão que pode ser tocado pelo usuário. Pode ser definido assim:

```
<element>
  <id>4</id>
  <type>3</type>
  <percentTop>40</percentTop>
  <percentLeft>0</percentLeft>
  <percentWidth>20</percentWidth>
  <percentHeight>10</percentHeight>
  <text>Menu</text>
  <onClickScript>lua::aamo.showMenu()</onClickScript>
</element>
```

É o controle de tipo 3.

Eventos

- “onClickScript”: invocado quando o usuário toca no botão;

Funções

Um BUTTON não possui funções de manipulação.

CHECKBOX

Uma checkbox é uma caixa de escolha binária (ligado / desligado). Seu tipo é 4 e pode ser definida assim:

```
<element>
  <id>3</id>
  <type>4</type>
  <percentTop>25</percentTop>
  <percentLeft>5</percentLeft>
  <percentWidth>10</percentWidth>
  <percentHeight>10</percentHeight>
  <checked>1</checked>
  <onChangeScript>trocou</onChangeScript>
</element>
```

A propriedade “checked” é um valor binário: 0 (zero) ou 1 (um), significando, respectivamente, desmarcada e marcada.

Eventos

- “onChangeScript”: invocado quando o status da CheckBox é alterado;

Funções

- estado = aamo.**getCheckBox**(<id da CHECKBOX>) : retorna 0 (zero) ou 1 (um), dependendo do estado da checkbox (desmarcada ou marcada, respectivamente);
- aamo.**setCheckBox**(<id da CHECKBOX>, <valor>): marca ou desmarca a checkbox, dependendo do valor passado (1 ou 0, respectivamente);

LISTBOX

Uma lista de elementos que permite ao usuário selecioná-los. Semelhante a um menu, porém mais flexível, pois pode ser alterada dinamicamente (e pode haver mais de uma Listbox por tela). Seu tipo é 5, e pode ser definida assim:

```
<element>
```



```
<id>1</id>
<type>5</type>
<percentTop>0</percentTop>
<percentLeft>0</percentLeft>
<percentWidth>100</percentWidth>
<percentHeight>40</percentHeight>
<onElementSelected>processar</onElementSelected>
</element>
```

Quando o usuário seleciona algum item da lista, o tratamento de evento “onElementSelected” é invocado e são criados dois parâmetros globais (veja a definição de parâmetro global mais adiante):

- “aamo::selectedIndex”: contém o índice selecionado (começando de zero);
- “aamo::selectedText”: contém o texto selecionado;

Eventos

- “onElementSelected”: ocorre quando o usuário toca em um item da lista. Você deve verificar os dois parâmetros globais para saber qual foi o elemento;

Funções

- **aamo.addListBoxOption**(<id da LISTBOX>, “<texto da opção>”): adiciona o texto informado à listbox indicada;
- **aamo.clearListBox**(<id da LISTBOX>): apaga todos os elementos da listbox indicada;

WEBBOX

Um mini-navegador web embutido na tela. Seu tipo é 6 e pode ser definido assim:

```
<element>
  <id>1</id>
  <type>6</type>
  <percentTop>0</percentTop>
  <percentLeft>0</percentLeft>
  <percentWidth>100</percentWidth>
  <percentHeight>40</percentHeight>
  <url>http://www.google.com</url>
</element>
```

O parâmetro “url” indica qual a URL inicial cuja página deverá ser exibida. Note que poderá ser traduzida também:

```
<url>I10n::urlInternacional</url>
```

Eventos

Atualmente a WEBBOX não responde a eventos.

Funções

- **aamo.navigateTo**(<id da WEBBOX>, "<texto da URL>"): faz com que a página seja mostrada na WEBBOX indicada. Pode ser traduzida também, por exemplo:
`aamo.navigateTo(1, "l10n::murl");`

IMAGEBOX

Uma caixa para exibir imagens. Seu tipo é 7 e pode ser declarada assim:

```
<element>
  <id>4</id>
  <type>7</type>
  <percentTop>75</percentTop>
  <percentLeft>0</percentLeft>
  <percentWidth>100</percentWidth>
  <percentHeight>30</percentHeight>
  <picture>l10n::imagem</picture>
  <stretch>1</stretch>
  <onClickScript>lua::aamo.showMessage("Clicou Imagem")</onClickScript>
</element>
```

Ela tem duas propriedades específicas:

- **picture**: nome do arquivo de imagem ou chave de tradução (pode ser internacionalizada);
- **stretch**: se 1 (um) a imagem é esticada para cobrir a IMAGEBOX. Se 0 (zero) a imagem é exibida em sua proporção normal;

Eventos

- **"onClickScript"**: caso o usuário toque na imagem, este tratamento é invocado;

Funções

- **aamo.setPicture**(<id da IMAGEBOX>, "<nome do arquivo>"): altera a imagem que está sendo exibida pela IMAGEBOX.

API de programação

O AAMO disponibiliza várias funções para uso dentro do código-fonte Lua.

Conforme já mencionado, o código pode ser especificado de maneira imediata, dentro dos tags de tratamento de evento (prefixo: "lua:."), ou então dentro de um arquivo externo, com extensão "lua".

Se houver erro no processamento de comandos Lua (sejam imediatos ou externos), mensagens de erro serão mostradas na console do projeto nativo. Com o AAMO Editor será possível executar e depurar o código Lua.

Todas as funções da API AAMO começam com o prefixo "aamo.". Vamos apresentar a lista em ordem alfabética.

addListBoxOption

- **aamo.addListBoxOption**(<id da LISTBOX>, "<texto da opção>"): adiciona o texto informado à listbox indicada;

clearListBox

- **aamo.clearListBox**(<id da LISTBOX>): apaga todos os elementos da listbox indicada;

close

- **aamo.closeCursor**("<nome da consulta>"): fecha o cursor de navegação de uma consulta ao banco de dados;

closeDatabase

- **aamo.closeDatabase**("<nome do banco de dados>"): fecha um banco de dados que foi aberto (ou criado). Pode ser utilizado quando a tela entra em pausa, para poupar recursos;

eof

- **estado = aamo.eof**("<nome da consulta>"): retorna o valor booleano "True" se acabaram os registros retornados pela consulta ao banco de dados. É necessário informar o nome da consulta;

execSQL

- `aamo.execSQL(<texto SQL>, <parametros SQL>)`: executa a query de atualização especificada no texto SQL, utilizando os parâmetros SQL passados. Exemplo:

```
aamo.execSQL("insert into contato (id , nome, endereco, email) values (?, ?, ?, ?)",  
NULL, aamo.getTextField(4),  
aamo.getTextField(6),  
aamo.getTextField(2))
```

exitScreen

- `aamo.exitScreen()`: descarrega a tela atual da pilha;

A tela atual será descarregada da pilha (e da memória). Antes disso, seu evento "onEndScript" será invocado. A tela imediatamente anterior (na pilha) será tornada atual, e seu evento "onBackScript" será executado.

Porém, se a tela atual for a primeira tela, o comando "exitScreen" não será executado.

getCheckBox

- `estado = aamo.getCheckBox(<id da CHECKBOX>)` : retorna 0 (zero) ou 1 (um), dependendo do estado da checkbox (desmarcada ou marcada, respectivamente);

getCurrentScreenId

- `id = aamo.getCurrentScreenId()`: retorna o identificador da tela atual;

getError

- `errorcode = aamo.getError()`: retorna o último código de erro AAMO;

Atualmente o AAMO suporta os seguintes códigos de erro:

- 10: Parâmetro faltando;
- 11: Arquivo não encontrado;
- 12: Valor igual a nulo;
- 100: Arquivo de localização não encontrado;

getGlobalParameter

- `<objeto> = aamo.getGlobalParameter("<nome>")`: retorna o valor do parâmetro cujo nome foi informado ou nil, caso não exista;

getLabelText

- `texto = aamo.getLabelText(<id do LABEL>)`: retorna o conteúdo do LABEL;

getLocalizedText

- `texto = aamo.getLocalizedText("<chave>")`: procura a chave no arquivo de tradução corrente e retorna sua tradução. Não use o prefixo "l10n::" no parâmetro desta função!

getTextField

- `texto = aamo.getTextField(<id da TEXTBOX>)`: obtém o conteúdo digitado em uma TEXTBOX;

loadScreen

- `aamo.loadScreen(<id da tela>)`: carrega uma tela na pilha;

Não importa se a tela já está ou não na pilha, pois uma nova tela será criada e carregada. O evento "onLoadScript" será invocado. Na tela anterior ocorrerá o "onLeaveScript". A nova tela será considerada a tela atual e seus controles e eventos estarão disponíveis para o usuário.

log

- `aamo.log("<texto>")`: grava uma mensagem no log do dispositivo;

navigateTo

- `aamo.navigateTo(<id da WEBBOX>, "<texto da URL>")`: faz com que a página seja mostrada na WEBBOX indicada. Pode ser traduzida também, por exemplo:
`aamo.navigateTo(1, "l10n::murl")`;

next

- `<tabela Lua> = aamo.next("<nome da consulta>")`: posiciona no próximo registro da consulta cujo nome foi passado. Este comando deve ser utilizado somente após um "aamo.query". O resultado é uma tabela de Strings com os números e conteúdos dos campos (somente Strings);

Nesta versão, devido às deficiências do acesso ao SQLite do Android 2.2, os valores dos campos são retornados em formato String.



openDatabase

- **aamo.openDatabase**("<nome do banco de dados>"): abre o banco de dados cujo nome está dentro do arquivo "bd/bd.xml" (tag: <name>);

Nesta versão do AAMO somente um único banco de dados pode ser aberto. Na verdade, só pode haver um único banco de dados (bd.xml dentro da subpasta "bd", dentro de "app"). O banco de dados é definido de acordo com o arquivo XML, caso não exista. Veja mais na parte sobre acesso a bancos de dados.

query

- **<tabela Lua> = aamo.query**("<nome da consulta>", "<comando SQL select>", "<parâmetros>"): cria uma nova consulta e retorna os campos do primeiro registro, como uma tabela Lua com Strings. Exemplo:

```
cursor = aamo.query("consulta","select * from contato where id = ?", id)
```

Leia mais sobre bancos de dados no capítulo específico.

setCheckBox

- **aamo.setCheckBox**(<id da CHECKBOX>, <valor>): marca ou desmarca a checkbox, dependendo do valor passado (1 ou 0, respectivamente);

setGlobalParameter

- **aamo.setGlobalParameter**("<nome>", <conteúdo>): adiciona um parâmetro global (que vale durante a execução da aplicação), com o nome informado e o conteúdo, que pode ser: texto, número ou boolean;

Um Parâmetro Global pode ser utilizado para armazenar informações que devem ser compartilhadas por toda a aplicação. O próprio AAMO faz uso dos parâmetros globais para indicar elementos de lista ou de menu selecionados. Se já existir um parâmetro com o nome informado, seu valor será substituído pelo novo conteúdo informado.

setLabelText

- **aamo.setLabelText**(<id do LABEL>, "<texto do label>"): altera o conteúdo do LABEL;

setPicture



- `aamo.setPicture(<id da IMAGEBOX>, "<nome do arquivo de imagem>")`: carrega a imagem indicada na imagebox. Note que o arquivo deve estar dentro da pasta "app".
Exemplo: `aamo.setPicture(1, "teste.png");`

setTextField

- `aamo.setTextField(<id da TEXTBOX>, "<Texto>")`: altera o conteúdo de uma TEXTBOX;

showMenu

- `aamo.showMenu()` : ao ser invocado exibe o menu configurado para a tela.

showMessage

- `aamo.showMessage("<texto da mensagem>")`: exibe uma mensagem para o usuário, e ele terá que tocar no botão "OK" do alerta;

showScreen

- `aamo.showScreen(<id da tela>)`: verifica se já existe uma tela com esse número na pilha. Se existir, promove para tela atual. Se não existir, carrega uma nova versão da tela;

Vai descarregando da pilha todas as telas, da atual até a tela desejada, invocando os eventos:

- "onEndScript": para cada tela descarregada;
- "onBackScript": só o da tela desejada;



Acesso a bancos de dados

O AAMO permite criar e acessar bancos de dados SQLite, de forma totalmente transparente e independente de plataforma. Ambas as plataformas atendidas pelo AAMO (Android e iOS) permitem o uso de bancos de dados SQLite.

Nesta versão há algumas limitações:

1. Somente pode ser declarado um único banco de dados;
2. Somente pode ser aberto um único banco de dados;
3. As consultas somente retornam o valor dos campos em formato String;

Para usar um banco de dados é necessário abri-lo, o que pode ser feito com o comando: `aamo.openDatabase("<nome do banco de dados>")`. Vamos ver um exemplo simples:

```
aamo.openDatabase("contatos")
cursor = aamo.query("consulta","select * from contato")
while not aamo.eof("consulta") do
    for i=1,#cursor do
        aamo.log(cursor[i])
    end
    cursor = aamo.next("consulta")
end
aamo.close("consulta")
aamo.closeDatabase("contatos")
```

Neste exemplo navegamos em todos os registros retornados, e em todos os campos de cada registro, exibindo no log os conteúdos. Note a condição “#cursor” no comando “for”, que permite navegar em todos os elementos de uma tabela Lua.

O comando `aamo.openDatabase` cria o banco de dados, caso não exista. Ele utiliza a definição armazenada em “app/bd/bd.xml”. Eis o arquivo de definição do banco de dados:

```
<?xml version="1.0" encoding="UTF-8"?>
<aamo-bd>
  <version>1</version>
  <name>contatos</name>

  <table>
    <name>contato</name>
    <columns>
      <column>
        <primarykey/>
        <name>ID</name>
        <type>integer</type>
        <notnull/>
      </column>
      <column>
        <name>nome</name>
        <type>text</type>
      </column>
    </columns>
  </table>
```



```
<column>
  <name>endereco</name>
  <type>text</type>
</column>
<column>
  <name>email</name>
  <type>text</type>
</column>
</columns>
</table>
</aamo-bd>
```

Mensagens de erro

O acesso a bancos de dados pode retornar 3 tipos de erro:

- 20: Erro ao abrir banco de dados;
- 21: Erro ao executar consulta (retorna nil);
- 22: Erro ao executar SQL (atualização);