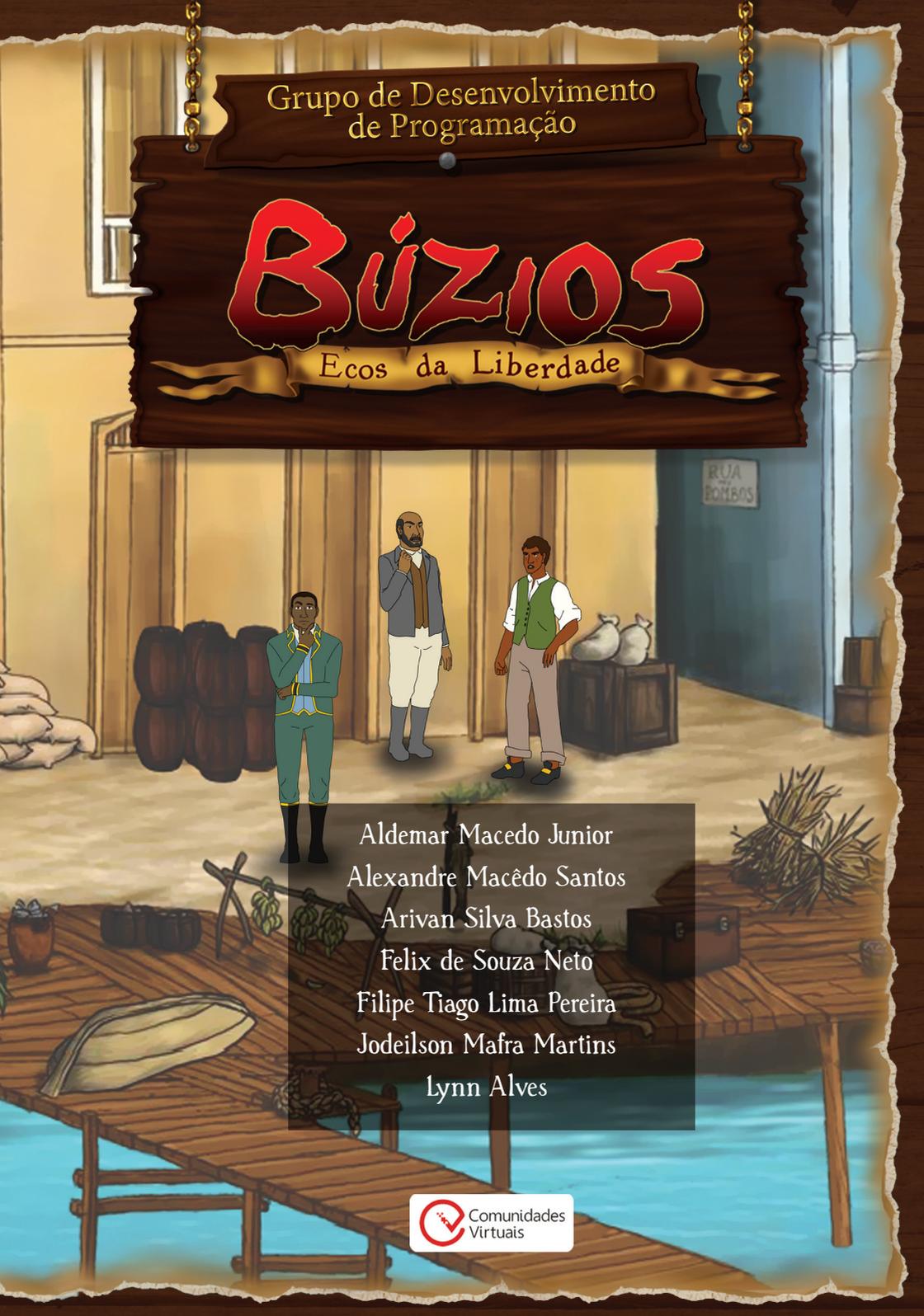


Grupo de Desenvolvimento
de Programação

Búzios

Ecossistema da Liberdade



Aldemar Macedo Junior
Alexandre Macêdo Santos
Arivan Silva Bastos
Felix de Souza Neto
Filipe Tiago Lima Pereira
Jodeilson Mafra Martins
Lynn Alves



Comunidades
Virtuais

O Búzios é um jogo adventure sobre a Revolta dos Búzios que aconteceu na Bahia no fim do século XVIII. Esse game foi desenvolvido na Universidade do Estado da Bahia com o financiamento da FAPESB.

O Projeto Búzios: Ecos da Liberdade tem o objetivo de simular o contexto da sociedade baiana no século XVIII, especificamente a Revolta dos Búzios, e resgatar uma revolta popular que se desenvolveu na Bahia, criando um espaço para que os alunos possam construir conceitos e significados acerca desse conteúdo histórico mediado por um jogo eletrônico.



Grupo de Desenvolvimento
de Programação

Búzios

Ecos da Liberdade

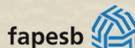


Distribuidor/
Desenvolvedor:



Comunidades
Virtuais

Financiamento:



Fundação de Amparo
à Pesquisa do Estado da Bahia



Secretaria de Ciência,
Tecnologia e Inovação

Apoio:



UNEB
UNIVERSIDADE DO
ESTADO DA BAHIA



Conselho Nacional de Desenvolvimento
Científico e Tecnológico





Grupo Gestor

Lourivaldo Valentim da Silva
Reitor

Amélia Tereza Santa Rosa Maraux
Vice-Reitora

José Bites de Carvalho
**Pró-Reitor de Ensino de Graduação
(PROGRAD)**

José Claudio Rocha
**Pró-Reitor de Pesquisa e Ensino de
Pós-Graduação (PPG)**

Adriana dos Santos Marmorini Lima
Pró-Reitora de Extensão (PROEX)

José Durval Uzêda Filho
**Pró-Reitor de Administração
(PROAD)**

Lídia Boaventura Pimenta
Chefe de Gabinete

Eduardo Lessa Guimarães
**Chefe da Procuradoria Jurídica
(PROJUR)**

Luiz Paulo de Almeida Neiva
**Responsável pela implantação do
PROPLAN**

Djalma Fiúza Almeida
**Chefe da Unidade de
Desenvolvimento Organizacional
(UDO)**

Paulo José Gonçalves de Souza
**Responsável pela implantação da
PRAES**

Wilson Roberto de Mattos
Zita Maria Farias Gomes Guimarães
Luiz Carlos dos Santos
Assessores Especiais

Antônio Amorim
**Diretor do Departamento de
Educação**

Eliseu Clementino Souza
**Coordenador do Programa de
Pós-graduação em Educação e
Contemporaneidade**

Equipe

Coordenação

Lynn Alves (*Coordenadora / Produtora*)

Josemar Rodrigues (*Vice-coordenador*)

Game Design

Aldemar Macedo Junior

Alexandre Macêdo Santos

Arivan Silva Bastos

Felix de Souza Neto

Filipe Tiago Lima Pereira

Jodeilson Mafra Martins

Lynn Alves

Raphael de Andrade Montenegro

Túiris Andrade de Azevedo

Artes e Design

Alexandre Macêdo Santos

André Luís Pereira dos Santos

Danilo Dias

Felix de Souza Neto

Jailce Fernanda Brito

Leomir Costa

Mariá de Sá Frizzera Scárdua

Piero Carapirá

Raphael de Andrade Montenegro

Raquel Meira de Andrade

Túiris Andrade de Azevedo

Vitor Torres

Roteiro

Clara Torres Peixoto

Filipe Tiago Lima Pereira

Isa Beatriz da Cruz Neves

Ivana Carolina Souza

Lynn Alves

Jodeilson Mafra Martins

Velda Gama Alves

Historiadores

Filipe Tiago Lima Pereira

Isa Beatriz da Cruz Neves

Jodeilson Mafra Martins

Trilha Sonora

Aldemar Macedo Junior

Programação

Arivan Silva Bastos

Laíza Costa Camurugy

Equipe Pedagógica

Alessandra Silva de Souza

Glória Araújo

Lygia Fuentes

Lynn Alves

Vanessa dos Santos Rios

Tatiana Santos da Paz

Consultoria

Cícero Bathomarco (*Roteiro*)

Jaci Maria Ferraz Menezes (*História*)

Revisora Linguística

Obdália Ferraz

Agradecimentos

Cícero Bathomarco

Eduardo Borges

Edyala Iglesia

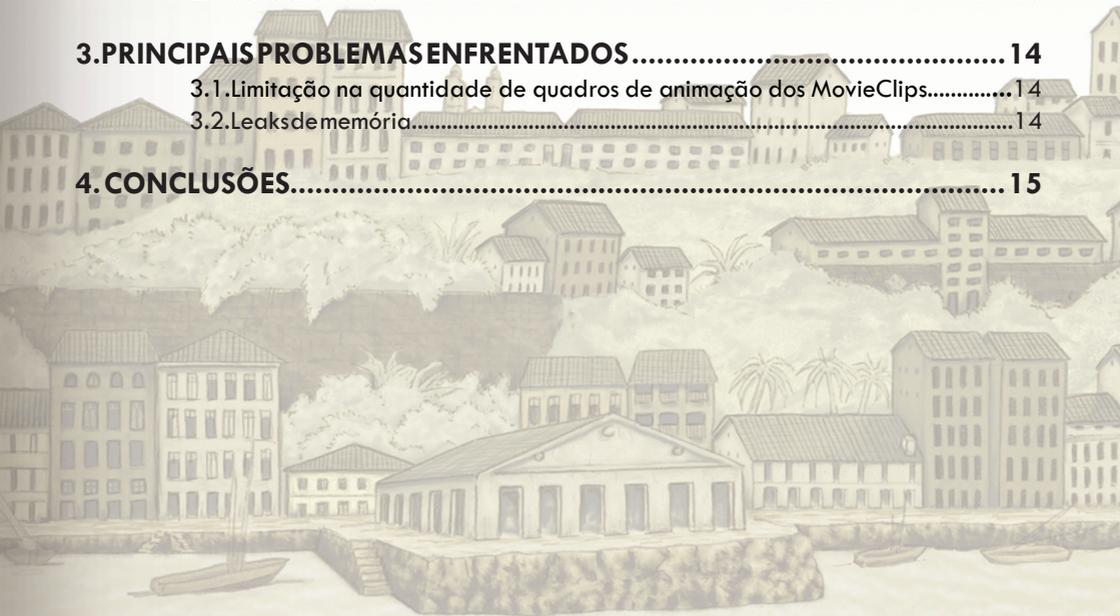
Jaci Maria Ferraz Menezes

Pola Ribeiro

Sofia Federico

Sumário

1.INTRODUÇÃO.....	08
1.1.Estilo Adventure.....	08
1.2 .Adobe Flash / Action Script 3.....	08
2.ARQUITETURA DO MOTOR.....	09
2.1.Roteiro.....	09
2.2.Localização de Rotas.....	10
2.3.Perspectiva da Cena	11
2.4.Manutenção do Estado do Jogo.....	12
2.5.Save and Load.....	13
3.PRINCIPAIS PROBLEMAS ENFRENTADOS.....	14
3.1.Limitação na quantidade de quadros de animação dos MovieClips.....	14
3.2.Leaks de memória.....	14
4. CONCLUSÕES.....	15





1. INTRODUÇÃO

Essa sessão descreve as principais questões ligadas programação do jogo Búzios.

1.1. O estilo Adventure

Jogos no estilo Adventure exigem um motor que ofereça os seguintes recursos:

1.3. Adobe Flash / Action Script 3

O Adobe Flash é uma tecnologia voltada para produção de aplicações multimídia interativas, para veiculação na Internet.

O Flash em conjunto com sua linguagem de programação, o Action Script, constitui uma ferramenta muito poderosa, pois reúne em um só ambiente ferramentas de desenho, animação e programação, com captura de eventos do usuário e funções de manipulação de símbolos, controle de colisão, desenho, entre outros.

Diante disso, o Flash vem sendo amplamente utilizado para produção de jogos eletrônicos. Entretanto, por se tratar de uma tecnologia voltada para veiculação na WEB, oferece algumas limitações que discutiremos ao longo desse memorial.

2.ARQUITETURA DO MOTOR

O motor do jogo Búzios, foi escrito em Action Script 3. Seus principais módulos são:

2.1.Roteiro

Todo o roteiro do jogo é definido através de arquivos XML. Existem três tipos de arquivos XML:

- Arquivo de definição de cena: definem características da cena (fase), como o arquivo .swf que a representa, suas dimensões, música de fundo, entre outros.
- Arquivo de definição de objeto: define as interações que o jogador pode realizar com aquele objeto e o que se procede ao realizar tais interações.
- Arquivo de definição de personagem: define as interações que o jogador pode realizar com aquele personagem, e o que se procede ao realizar tais interações.

Os arquivos XML que compõem o roteiro encontram-se dentro da pasta “data” do jogo, obedecendo a seguinte estrutura de diretórios:

```
data\nome-da-fase\cena.xml
```

```
data\nome-da-fase\objetos\nome-do-objeto.xml
```

```
data\nome-da-fase\personagens\nome-do-personagem.xml
```

Os arquivos de definição de objeto e personagem, são divididos em duas partes: definição de diálogos e definição de interações. A especificação de diálogos define todos os diálogos que podem ser realizados com aquele personagem ou objeto.

Interações podem ser de quatro tipos: pegar, observar, falar ou combinar. Cada definição de interação é constituída de condições e ações. Condições determinam as exigências para aquela interação ser executada. As ações determinam o que será executado diante daquela interação.

Condições	
Condição	Descrição
Variável	Verifica se a variável possui um determinado valor
Ações	
Ação	Descrição
variavel	Define o valor de uma variável.
mudar_cena	Transporta o jogador para um outro cenário.
iniciar_animacao	Dispara uma animação em um personagem
Iniciar_dialogo	Inicia um diálogo.
adicionar_inventario	Adiciona um objeto ao inventário do jogador.
minigame	Inicia um minigame.

Tabela 1: exemplos de condições e ações.

2.2. Localização de rotas

Por não ser uma ferramenta voltada especificamente para o desenvolvimento de jogos, o Flash não trás recursos de localização de rotas. Para implantar a localização de rotas, foi criado um sistema de mapeamento, onde as áreas caminháveis do cenário são definidas por um MovieClip, e, a partir dele, cria-se uma matriz de colisão, usada como base para um algoritmo de localização de rotas A*.



Figura 1: cenário do navio



Figura 2: cenário do navio com destaque para a região caminhável.

Y do personagem e objetos são sempre o ponto de registro, que precisa ser posicionado adequadamente para que o sistema funcione.

Assim, o sistema de movimentação do personagem troca as profundidades do personagem e dos objetos, de acordo com sua posição no cenário, usando para isso a função `SwapChildren()` oferecida pelo Flash. Tal função recebe como parâmetros dois elementos, trocando a posição Z entre eles.

Por fim, o personagem é escalonado (redimensionado) a medida que se afasta ou se aproxima na perspectiva da cena, criando a sensação de que está caminhando no cenário e se afastando de uma câmera existente. Para redimensioná-lo, são utilizadas as propriedades `scaleX` e `scaleY` do `MovieClip` do personagem, que permitem definir o tamanho de um elemento em de acordo com uma percentagem em relação ao seu tamanho original.

2.4. Manutenção do estado do jogo

Durante a execução do jogo as ações realizadas pelo jogador podem significar alterações no estado do jogo. Se, por exemplo, o jogador realizar uma interação com o objeto 1 no cenário A, interação esta que cause uma mudança permanente no referido objeto, tal mudança precisa ser registrada, pois caso o jogador volte ao cenário A, o objeto 1 deve apresentar-se como fora deixado.

Por outro lado, algumas ações não são permanentes. Por exemplo, se o jogador aciona a abertura da porta de um armário e o personagem examina o interior do armário mas em seguida fecha novamente a porta, nenhuma mudança de estado precisa ser registrada.

Dessa forma, não cabe ao motor registrar todas as interações como mudanças de estado ou mesmo decidir qual mudança de estado deve ser registrada. No motor do Búzios, os registros de estado são especificados no roteiro, através da ação do tipo `VARIAVEL` e da condição do tipo `VARIAVEL`.

Se o jogador alterar o estado de um objeto, uma variável que represente o estado daquele objeto deverá ser gravada. No roteiro de definição da fase, no segmento de inicialização, haverá um conjunto de ações associados a cada valor daquela variável, permitindo iniciar adequadamente o objeto de acordo com o valor que a variável possui.

Por exemplo, se ao interagir da forma `PEGAR` com um interruptor localizado no cenário A, o interruptor for ligado, uma variável de nome `"CENARIOA.INTERRUPTOR.ESTADO"` pode assumir o valor `"LIGADO"`. Na inicialização do cenário, podemos verificar o valor da variável `"CENARIOA.INTERRUPTOR.ESTADO"`, realizando as ações apropriadas como por

exemplo atualizar o símbolo gráfico que representa o interruptor e ligar a luz correspondente.

2.5. Salve e Load

O mecanismo de salvamento e carregamento do motor usa como base o sistema de manutenção de estado do jogo. Todas as variáveis são gravadas no arquivo de save. Algumas outras informações adicionais também são necessárias, como: cenário em que o personagem se encontra, posição do personagem no cenário, quadros atuais dos elementos que estão sendo animados no cenário, entre outros.

Diante das restrições de segurança impostas por uma tecnologia voltada para a WEB, o flash não oferece recursos e leitura e gravação em disco. Para realizar tais funções foi necessário utilizar a extensão Zinc, que oferece recursos adicionais ao Flash. Diante



3. PRINCIPAIS PROBLEMAS ENFRENTADOS

Durante o processo produtivo do Búzios foram encontradas duas principais dificuldades:

3.1. Limitação na quantidade de quadros de animação dos MovieClips

Em um jogo Adventure, quanto mais elementos interativos um cenário possuir, maior a sensação de liberdade experimentada pelo jogador. Em um adventure ideal, todos objetos no cenário poderiam ser observados e descritos pelo personagem, assim como manipulados, o que não ocorre devido aos limites de tempo e recursos de desenvolvimento.

A necessidade de criar um ambiente rico em objetos interativos e animações foi dificultada pelos limites de memória impostos no Adobe Flash CS4. Em cenários com muitos objetos animados, o programa demorava muito em responder durante a manipulação dos elementos e travava com frequência, mesmo após aplicadas todas as atualizações disponíveis.

Foi necessário simplificar e reduzir a quantidade de animações, bem como o tamanho de alguns cenários. Quanto mais objetos, animações e quanto maiores eram os movieclips, mais o Flash ficava lento e instável.

3.2. Leaks de memória

Um outro problema observado ao desenvolver um jogo no Flash, é a facilidade que essa tecnologia tem em criar “leaks” de memória. Em um motor para jogos Adventure, é necessário carregar e descarregar cenários da memória com frequência, pois o jogador tem grande liberdade em navegar pelo cenário. Ao carregar um cenário é necessário criar-se alguns event listeners nele, para, por exemplo, capturar os cliques do jogador sobre os objetos.

Nesse aspecto, o principal problema do Action Script 3 é que ele não oferece nenhum recurso para forçar a remoção completa de um filme (no caso o cenário) da memória da aplicação em execução. Caso um filme seja removido e exista alguma referência que ainda aponte para algum elemento dele, o filme continuará na memória.

Para evitar leaks de memória no flash é preciso assegurar-se que:

- Nenhuma variável apontando para algum elemento do filme que deseja-se descarregar.
- Todos “event listeners” criados foram destruídos.

4.CONCLUSÕES

O desempenho durante a execução do jogo Búzios bem como as limitações na quantidade de objetos durante a criação dos cenários, demonstram, em uma análise inicial, que o Flash não oferece os recursos necessários para a criação de um jogo estilo Adventure com as características do Búzios. Para obter-se um grau de desempenho satisfatório, é necessário estudar alternativas para otimizar seu uso de memória, tanto no aspecto de edição dos cenários do jogo, como de execução dele.

O uso da extensão Zinc foi fundamental no projeto, pois permitiu a utilização de funções de acesso a arquivos em disco e facilitou o processo de publicação do jogo em formato executável para Windows e Linux.





Búzios

Ecos da Liberdade

Distribuidor/
Desenvolvedor:



Comunidades
Virtuais

Financiamento:

fapesb 
Fundação de Amparo
à Pesquisa do Estado da Bahia

 **Bahia**
TERRA DE TODOS NÓS
Secretaria de Ciência,
Tecnologia e Inovação

Apoio:


UNEb
UNIVERSIDADE DO
ESTADO DA BAHIA


Conselho Nacional de Desenvolvimento
Científico e Tecnológico

www.comunidadesvirtuais.pro.br/buzios