



UNIVERSIDADE LUSÓFONA DE HUMANIDADES E TECNOLOGIAS

PROJECTO FINAL DE CURSO

iEarNotes Aplicação móvel para treinar o ouvido musical

Tiago Daniel Santos Brás
20061881
Universidade Lusófona de
Humanidades e Tecnologias
Lisboa, Portugal
tiagobraspt@gmail.com

Orientador: Professor Mestre Alexandre Pereira

Setembro 24, 2010

Conteúdos

Resumo	ii
Abstract	iii
1. Introdução	1
2. Enquadramento Teórico	2
Acordes	2
Padrões de dedilhados	3
3. Método	4
Requisitos Funcionais	4
Requisitos Não-Funcionais	5
Casos de Uso	5
Especificação	9
Hierarquia de Menus	9
Sistema de troféus	9
Dados estatísticos	10
Quizzes	10
4. Resultados	11
Testes	11
5. Conclusão e trabalho futuro	14
Bibliografia	15
Anexos	16
Apêndice A - Teoria da música	17
Apêndice B - Diagrama de classes	17
Apêndice C - Fluxogramas	20
Apêndice D - Mapa de navegação	22
Apêndice E - <i>Layouts</i>	23
Apêndice F - Capturas de Ecrãs	24
Glossário	27

Resumo

A capacidade de distinguir e reconhecer acordes, notas e padrões de dedilhados de guitarra é uma das melhores qualidades que um guitarrista pode possuir. Além de aumentar a criatividade e sensibilidade musical do guitarrista também lhe permite transcrever músicas recorrendo somente à audição. Este trabalho tem assim como objectivo o desenvolvimento de uma aplicação simples e móvel para iPhone e iPod com esse mesmo fim. Ao contrário das aplicações disponíveis no mercado, esta, além de implementar técnicas que motivam e estimulam a aprendizagem, não parte do pressuposto que o utilizador já tem um ouvido musical minimamente desenvolvido. O que isto significa é que, nenhum acorde, nota ou padrão de dedilhado é avaliado antes de ser introduzido ao utilizador; introdução esta que inclui o som, notação e formação do mesmo. Por conseguinte, o utilizador consegue melhorar o seu ouvido musical mais rapidamente do que pelos meios convencionais.

Abstract

The ability to distinguish and recognize chords, musical notes and strumming patterns of the guitar is one of the most important skills that a guitar player may possess. Besides increasing the creativity and sensibility of the player, it also enables him to transcribe music by ear. Therefore, the objective of this work is to develop a mobile and easy-to-use application for the iPhone and iPod with that purpose in mind. Unlike most of the applications available in the market, this one, in addition to the implementation of stimulating and motivating learning techniques, does not assume that the user has already a minimally developed musical ear. What this means is, no chord, musical note or strumming pattern is tested before being introduced to the user by a brief introduction screen that includes its audio, notation and formation. As a result, the user is able to improve his musical ear quicker than by conventional means.

Keywords: ear training, transcription, music, chords, iPhone.

Introdução

O objectivo deste trabalho é criar uma aplicação fácil de usar que ajude a treinar o ouvido musical.

O público alvo principal são todos aqueles que tocam guitarra, independentemente do nível, e que desejam melhorar a sua capacidade de identificar acordes, notas e padrões de dedilhados pela audição. O treino, propriamente dito, é feito respondendo a várias perguntas sobre a área que deseja melhorar. O tipo de perguntas mais habituais são aquelas em que é dado o som de uma nota, acorde ou padrão de dedilhado e o utilizador tem de escolher a resposta correcta entre várias opções.

As características que fazem com que esta aplicação se distinga das similares são: acompanhamento personalizado a cada indivíduo, troféus e indicador de progresso. Acompanhamento personalizado significa que o programa molda-se ao utilizador consoante o seu histórico de respostas, conseguindo assim identificar as dificuldades do utilizador em questão. Troféus são objectivos predefinidos com o intuito de motivar o utilizador a atingir determinadas metas, por exemplo, acertar correctamente a 10 perguntas de seguida. Indicador de progresso consiste em quantificar o progresso do utilizador num valor numérico.

A plataforma escolhida foi o iPhone/iPod, com sistema operativo iOS 4.0 ou superior. As razões de escolha foram: ser um dispositivo móvel, ou seja, a pessoa pode treinar em qualquer lugar; ser fácil de usar; possuir uma interface indicada a este tipo de aplicação e o facto de deter uma enorme quota de mercado.

Enquadramento Teórico

Treinar o ouvido musical significa aprender a distinguir e a identificar os mais variados elementos da música, desde os acordes até aos padrões de dedilhados. Possuir um ouvido musical desenvolvido é muito importante para qualquer pessoa que toque um instrumento musical, já que, ao identificar tons permite reproduzi-los pela própria voz ou num instrumento. Para além disso, aumenta a sensibilidade que a pessoa tem da música, melhora a capacidade de isolar os vários instrumentos que constituem uma peça musical, permite transcrever o que se ouve, facilita e melhora a composição de músicas.

Pode-se dividir o reconhecimento de tons em duas categorias: reconhecimento absoluto e reconhecimento relativo. No reconhecimento absoluto o indivíduo consegue identificar com precisão o tom de uma nota individual enquanto que no reconhecimento relativo o indivíduo consegue identificar se uma nota é aguda ou grave em relação a outra e o seu intervalo. Ambas são importantes para quem toca um instrumento musical.

Este trabalho vai-se focar mais na identificação de acordes e padrões de dedilhados.

Acordes

Existem 12 notas na música ocidental; separadas por um semitom. Neste trabalho a notação do solfejo usado é a inglesa.

Um intervalo de um semitom corresponde a um trasto na guitarra. Uma guitarra acústica tem normalmente 19 a 24 trastos e seis cordas com a afinação "e B G D A E" da corda mais aguda para a corda mais grave.

O tipo mais simples de acorde é o tríade. Os tríades básicos que existem são:

- Tríade maior:** formado pela nota base do acorde, terça maior e quinta justa.
- Tríade menor:** formado pela nota base do acorde, terça menor e quinta justa.
- Tríade aumentado:** formado pela nota base do acorde, terça maior e quinta aumentada.
- Tríade diminuta:** formado pela nota base do acorde, terça menor e quinta diminuta.

Portanto os tríades básicos do acorde C são:

Nome	Nota base	Primeira componente	Segunda componente
C maior	C	E	G
C menor	C	E \flat	G
C aumentado	C	E	G \sharp
C diminuto	C	E \flat	G \flat

Para além de tríades também existem acordes de sétima menor, *power chords*, acordes suspensos, entre outros.

Padrões de dedilhados

Padrão de dedilhado é o nome que se dá à sequência repetitiva em que uma ou mais cordas da guitarra são tocadas.

As componentes mais usadas nos padrões de dedilhados são: *Downstroke* (D), *Upstroke* (U) e Arpejo.

O padrão mais simples é realizar uma *downstroke* em cada batida ou semínima. Num compasso 4/4 podemos representa-lo da seguinte forma:

D	D	D	D
↓	↓	↓	↓

Se dividirmos o compasso em oito unidades, podemos adicionar uma *upstroke* a cada *downstroke* e obtemos outro padrão.

DU	DU	DU	DU
↓↑	↓↑	↓↑	↓↑

Ou seja, por cada batida realiza-se uma *downstroke* e uma *upstroke*. Com isto, podemos criar mais padrões de dedilhados omitindo *downstrokes* e *upstrokes* específicos. Por exemplo, um dos padrões de dedilhados mais popular é:

D	DU	U	D
↓	↓↑	↑	↓

Portanto, dada a sequência de acordes G–Em–C–D podemos tocar da seguinte maneira:

G				Em				C				D			
D	DU	U	D	D	DU	U	D	D	DU	U	D	D	DU	U	D
↓	↓↑	↑	↓	↓	↓↑	↑	↓	↓	↓↑	↑	↓	↓	↓↑	↑	↓

Método

Requisitos Funcionais

- A aplicação tem como objectivo “treinar” o ouvido do utilizador para reconhecer e distinguir notas, acordes e padrões de dedilhados de guitarra.
- Existem dois modos para cada unidade de aprendizagem (acorde, notas, dedilhados): modo acompanhado e modo personalizado. Modo acompanhado como o nome indica, a aprendizagem é feita de um modo automatizado e os objectos de aprendizagem são introduzidos consoante o progresso do utilizador. No modo personalizado o utilizador decide o que quer treinar.
- A aprendizagem é feita através da realização de *quizzes* variados.
- Deve ser possível o utilizador proteger a sua conta com palavra passe.
- Deve ser permitido ignorar o *login* se o utilizador assim o desejar.
- A aplicação deve implementar um sistema de troféus para cada utilizador. Este sistema funciona como agente motivador, estimulando o utilizador a progredir. Basicamente, é um sistema que dá troféus aos utilizadores por estes alcançarem vários objectivos. Desde “Responder a 10 questões correctamente de seguida” a “Dominar 5 acordes”.
- O utilizador deve poder consultar os seus dados estatísticos referente a cada modo e unidade de aprendizagem. Por exemplo: “Acorde que domina mais”, “Quanto tempo gasto a treinar acordes”, “Quantos *quizzes* realizados”, etc.
- Deve ser permitido ao utilizador repor todos os dados de origem. Para tal, o utilizador tem de inserir, obrigatoriamente, a sua palavra chave.
- Internamente, o algoritmo de criação de *quizzes* deve implementar vários níveis de dificuldade e vários tipos de questões para que a experiência seja mais divertida e estimulante.
- No modo acompanhado o programa deve ter em conta a possível frustração do utilizador. Isto é, se verificar que o utilizador está a errar demasiadas vezes seguidas, deve baixar o nível de dificuldade dinamicamente e com isto fazer uma ou outra questão sobre objectos já dominados ou aprendidos.
- As questões devem permitir um número variado de respostas.
- Grande parte das questões devem ter um tempo limite de resposta. Este é dinâmico e tem em conta o progresso do utilizador. Por exemplo, ao início o utilizador pode ter mais tempo para responder.
- Devem ser implementados pelo menos os seguintes tipos de questões (para efeitos de simplicidade, serão usados os acordes como os objectos de aprendizagem):
 - É dado a notação do acorde e o utilizador tem de escolher entre várias opções o som correcto do acorde dado. Cada opção representa o som de um acorde.
 - É dado o som de um acorde e o utilizador tem de escolher entre várias opções a notação correcta do acorde dado. Cada opção representa a notação de um acorde.
 - São tocados mais do que um acorde e o utilizador tem de escolher entre várias opções as notações correctas dos acordes dados, pela ordem correcta. Cada opção representa a notação de um acorde.
- Deve haver um indicador de progresso para cada utilizador. Este deve representar o progresso do utilizador de uma forma imediata. Com isto, terá de ser implementado um sistema de pontuação específico para cada área de aprendizagem e/ou geral que atribui uma pontuação a cada resposta dada. Esta pontuação pode ter em conta vários parâmetros, tais como, tempo de resposta, dificuldade da questão e pontuação total do utilizador.
- Deverá ser possível mudar o tipo de guitarra.

Requisitos Não-Funcionais

- A aplicação deve ser desenvolvida para iPhone e iPod para o sistema operativo iOS 4.0 ou superior.
- Deve permitir até três utilizadores usarem a aplicação.
- Todos os dados e respostas são guardadas de forma permanente e associadas à conta do respectivo utilizador.
- A aplicação deve ser implementada de modo a ser facilmente localizada para várias línguas.
- A aplicação deverá ser fácil de usar.

Casos de Uso

Nome do Caso de Uso:	Criar uma nova conta de utilizador	
Descrição:	O caso de uso descreve o processo de criação de uma conta pelo utilizador.	
Actores:	Cliente.	
Precondição:	Nenhuma.	
Trigger	O caso de uso é iniciado quando o utilizador arranca a aplicação.	
Fluxo básico:	Acção do actor:	Resposta do sistema:
	Passo 1: O utilizador arranca a aplicação.	
	Passo 2: O utilizador toca no botão que simboliza “Utilizadores”	
	Passo 3: O utilizador escolhe “Criar nova conta”	
	Passo 4: O utilizador insere o nome do utilizador e a palavra chave.	
		Passo 5: O sistema verifica se o nome de utilizador já não existe e se as palavras chave combinam.
		Passo 6: A conta é criada.
Fluxo alternativo:	Passo 7: O menu principal da aplicação é mostrado.	
	Passo alternativo 2: Se não existir ainda nenhuma conta no sistema, o passo 2 é ignorado e salta para o passo 3.	
Conclusão:	Passo alternativo 6: Se o nome do utilizador já existir ou se as palavras chave não combinarem então o sistema mostra uma mensagem de erro e pede ao utilizador para inserir novamente os dados.	
	O caso de uso é concluído quando a conta é criada.	

Nome do Caso de Uso:	Repor todos os dados.	
Descrição:	O caso de uso descreve o processo de repor todos os dados do utilizador, tais como, troféus, pontos e dados estatísticos.	
Actores:	Cliente.	
Precondição:	Estar <i>logged in</i>	
Trigger	O caso de usar começa depois do utilizador se autenticar.	
Fluxo básico:	Acção do actor:	Resposta do sistema:
	Passo 1: O utilizador acede às opções.	
	Passo 2: O utilizador escolhe a opção “Repor dados”.	
		Passo 3: O sistema pergunta ao utilizador se deseja, de facto, repor todos os seus dados.
	Passo 4: O utilizador confirma que sim.	
		Passo 5: O sistema repõe todos os dados do utilizador.
Fluxo alternativo:	Passo alternativo 4: Se o utilizador cancelar a operação, os dados não são repostos.	
Conclusão:	O caso de uso é concluído quando todos os dados forem repostos ou se o utilizador cancelar a operação.	

Nome do Caso de Uso:	Distinguir o som de um acorde no modo acompanhado.	
Descrição:	O caso de uso descreve o processo em que é dado ao utilizador para distinguir/reconhecer o som de um acorde. Quando a questão é apresentada ao utilizador, o som do acorde é tocado e a contagem decrescente para responder à pergunta é iniciada. O utilizador tem ao seu dispor várias opções, por onde escolher, com a notação por extenso. O som do acorde pode ser tocado sempre que o desejar.	
Actores:	Cliente.	
Precondição:	Estar <i>logged in</i> .	
Trigger	O caso de usar começa depois do utilizador se autenticar.	
Fluxo básico:	Acção do actor:	Resposta do sistema:
	<p>Passo 1: O utilizador acede ao menu “Acordes”.</p> <p>Passo 2: O utilizado escolhe “Modo Acompanhado”</p> <p>Passo 3: O sistema cria uma questão tendo em conta o progresso do utilizador.</p> <p>Passo 4: A questão é mostrada ao utilizador, e o som do acorde é tocado.</p> <p>Passo 5: O utilizador escolhe uma das opções disponíveis.</p> <p>Passo 6: O sistema valida a resposta e calcula o número de pontos que o utilizador ganhou.</p> <p>Passo 7: O sistema mostra ao utilizador a resposta correcta e o número de pontos atribuídos.</p> <p>Passo 8: O sistema actualiza vários dados estatísticos.</p>	
Fluxo alternativo:	Passo alternativo 4: Se for a primeira vez que o acorde em questão aparece ao utilizador, o sistema fará uma pequena introdução ao novo acorde composta pelos seguintes dados: som, notação e composição.	
Conclusão:	O caso de uso é concluído quando o sistema actualiza todos os dados estatísticos.	

Nome do Caso de Uso:	Consultar troféus obtidos	
Descrição:	O caso de uso descreve o processo de consultar os troféus obtidos pelo utilizador até à data.	
Actores:	Cliente.	
Precondição:	Estar <i>logged in</i> .	
Trigger	O caso de usar começa depois do utilizador se autenticar.	
Fluxo básico:	Acção do actor:	Resposta do sistema:
	Passo 1: O utilizador escolhe a opção "Troféus" do menu principal.	
		Passo 2: O sistema mostra ao utilizador os troféus obtidos até ao momento.
Fluxo alternativo:	Nenhum.	
Conclusão:	O caso de uso é concluído assim que é mostrado os troféus ao utilizador.	

Especificação

Hierarquia de Menus

O menu principal da aplicação permite o utilizador ir para os seguintes ecrãs:

- Menu dos acordes
- Menu das notas
- Menu dos padrões de dedilhados
- Opções
- Utilizadores
- Troféus
- Dados estatísticos

Nos menus dos acordes, notas e padrões de dedilhados ainda temos as opções modo acompanhado e modo personalizado.

Sistema de troféus

Os troféus têm os seguintes atributos: nome opcional, número de pontos atribuídos pelo troféu e descrição do objectivo a alcançar.

- Tempo total gasto a usar a aplicação: 5, 20 horas. **5, 10 pontos**, respectivamente.
- Tempo total gasto no modo acompanhado: 3, 10 horas. **5, 10 pontos**, respectivamente.
- Tempo total gasto no modo personalizado: 5, 15 horas. **10, 20 pontos**, respectivamente.
- Tempo total gasto a treinar acordes: 1, 5 horas. **1, 5 pontos**, respectivamente.
- Tempo total gasto a treinar notas: 1, 5 horas. **1, 5 pontos**, respectivamente.
- Tempo total gasto a treinar padrões de dedilhados: 1, 5 horas. **1, 5 pontos**, respectivamente.
- Responder a 10, 100, 500, 1500 questões (modo acompanhado). **1, 5, 10, 15 pontos**, respectivamente.
- Responder correctamente a 1, 50, 250, 500, 1000 questões (modo acompanhado). **1, 5, 10, 15, 30 pontos**, respectivamente.
- Aprender 1, 5, 15, 30, 50 acordes (modo acompanhado). **1, 5, 10, 15, 20 pontos**, respectivamente.
- Dominar 1, 5, 15, 30, 50 acordes (modo acompanhado). **5, 15, 30, 50, 70 pontos**, respectivamente.
- Responder correctamente a 1, 50, 250, 500, 1000 questões sobre acordes (modo acompanhado). **1, 5, 10, 15, 30 pontos**, respectivamente.
- Responder correctamente a 3, 5, 10, 15, 25 questões sobre acordes de seguida sem falhar nenhuma (modo acompanhado). **5, 10, 15, 20, 30 pontos**, respectivamente.
- Aprender 1, 3, 5, 10, 12 notas (modo acompanhado). **1, 5, 10, 15, 20 pontos**, respectivamente.
- Dominar 1, 3, 5, 10, 12 notas (modo acompanhado). **5, 15, 30, 50, 70 pontos**, respectivamente.
- Responder correctamente a 1, 50, 250, 500, 1000 questões sobre notas (modo acompanhado). **1, 5, 10, 15, 30 pontos**, respectivamente.
- Responder correctamente a 3, 5, 10, 15, 25 questões sobre notas de seguida sem falhar nenhuma (modo acompanhado). **5, 10, 15, 20, 30 pontos**, respectivamente.
- Aprender 1, 5, 10, 20, 30 padrões de dedilhados (modo acompanhado). **1, 5, 10, 15, 20 pontos**, respectivamente.
- Dominar 1, 5, 10, 20, 30 padrões de dedilhados (modo acompanhado). **5, 15, 30, 50, 70 pontos**, respectivamente.
- Responder correctamente a 1, 50, 250, 500, 1000 questões sobre padrões de dedilhados (modo acompanhado). **1, 5, 10, 15, 30 pontos**, respectivamente.
- Responder correctamente a 3, 5, 10, 15, 25 questões sobre padrões de dedilhados de seguida sem falhar nenhuma (modo acompanhado). **5, 10, 15, 20, 30 pontos**, respectivamente.

Dados estatísticos

- Tempo total gasto a usar a aplicação.
- Tempo total gasto no modo acompanhado.
- Tempo total gasto no modo personalizado.
- Número total de questões realizadas no modo acompanhado.
- Número total de questões realizadas no modo personalizado.
- Tempo total gasto a treinar acordes (modo acompanhado).
- Tempo total gasto a treinar notas (modo acompanhado).
- Tempo total gasto a treinar padrões de dedilhados (modo acompanhado).
- Número de respostas correctas dadas ao treinar acordes (modo acompanhado).
- Número de respostas correctas dadas ao treinar notas (modo acompanhado).
- Número de respostas correctas dadas ao treinar padrões de dedilhados (modo acompanhado).
- Recorde de número de respostas correctas dadas de seguida ao treinar acordes (modo acompanhado).
- Recorde de número de respostas correctas dadas de seguida ao treinar notas (modo acompanhado).
- Recorde de número de respostas correctas dadas de seguida ao treinar padrões de dedilhados (modo acompanhado).
- Tempo total gasto a treinar padrões de dedilhados (modo acompanhado).
- Número de respostas erradas dadas ao treinar acordes (modo acompanhado).
- Número de respostas erradas dadas ao treinar notas (modo acompanhado).
- Número de respostas erradas dadas ao treinar padrões de dedilhados (modo acompanhado).
- Recorde de número de erradas correctas dadas de seguida ao treinar acordes (modo acompanhado).
- Recorde de número de erradas correctas dadas de seguida ao treinar notas (modo acompanhado).
- Recorde de número de erradas correctas dadas de seguida ao treinar padrões de dedilhados (modo acompanhado).
- Acorde mais fácil.
- Acorde mais difícil.
- Nota mais fácil.
- Nota mais difícil.
- Padrão de dedilhado mais fácil.
- Padrão de dedilhado mais difícil.

Quizzes

O tipo de questões possíveis são:

Nr.	Tipo de questão	Tipo de resposta	Método de resposta
1	Único som	Notação	Único
2	Vários sons em sequência	Notação	Sequencial
3	Notação	Som	Único

Exemplos:

Tipo 1: É tocado o som de um acorde e o utilizador tem de escolher a notação correcta do acorde entre várias possibilidades.

Tipo 2: São tocados vários acordes em sequência e o utilizador tem de escolher as notações correctas de seguida entre as várias possibilidades.

Tipo 3: É dada a notação de um acorde e o utilizador tem de escolher entre o som correcto do acorde entre várias possibilidades.

Resultados

Testes

Nome do teste:	Serialização da classe 'User'.
Descrição:	Testar se todos atributos da classe 'User' são correctamente serializados.
Método:	<p>Criar um <i>user</i>, serializa-lo, grava-lo num ficheiro, desalojar a memória ocupada pelo <i>user</i>, e por fim, de-serializar o <i>user</i> a partir do ficheiro.</p> <p>Código do teste:</p> <pre>User *user = [[User alloc] initWithDefaultUser]; // Serializar NSData *data = [NSKeyedArchiver ↵ archivedDataWithRootObject:user]; [data writeToFile:@"user.dat" atomically:YES]; [data release]; data = nil; [user release]; user = nil; data = [NSData dataWithContentsOfFile:@"user.dat"↵]; // De-serializar user = [[NSKeyedUnarchiver unarchiveObjectWithData↵ :data] retain]; [data release]; // Testar a integridade dos dados do 'user' no ↵ debugger [user release];</pre>
Resultado esperado:	Que os atributos do <i>user</i> depois da de-serialização sejam os mesmos que antes da serialização.
Resultado:	Teste concluído com sucesso.

Nome do teste:	Aleatoriedade condicionada.
Descrição:	<p>Testar o comportamento da função implementada em 'NSMutableArray+Random.m':</p> <pre>-(id)randomObjectExceptAnyObjectFrom: (NSMutableArray *)array;</pre> <p>que consiste em seleccionar um elemento aleatoriamente do próprio vector que não pertença a <i>array</i>.</p>
Método:	<p>Dados os seguintes dois vectores de inteiros, $v = (0, 1, 2, 3, 4, 5)$ e $u = (2, 4)$, correr a função "randomObjectExceptAnyObjectFrom" 1000 vezes e verificar se os valores retornados não pertencem de facto ao vector u.</p> <p>Código do teste:</p> <pre>NSNumber *n0 = [NSNumber numberWithInt: 0]; NSNumber *n1 = [NSNumber numberWithInt: 1]; NSNumber *n2 = [NSNumber numberWithInt: 2]; NSNumber *n3 = [NSNumber numberWithInt: 3]; NSNumber *n4 = [NSNumber numberWithInt: 4]; NSNumber *n5 = [NSNumber numberWithInt: 5]; NSMutableArray *v = [[NSMutableArray alloc] initWithObjects: n0, n1, n2, n3, n4, n5↵ , nil]; NSMutableArray *u = [[NSMutableArray alloc] initWithObjects: n2, n4, nil]; int counters[6] = {0, 0, 0, 0, 0, 0}; for (int i = 0; i < 1000; i++) { counters [[v randomObjectExceptAnyObjectFrom:u] ↵ intValue]]++; } [v release]; [u release]; NSLog(@"Nr. de 0s: %d\n " "Nr. de 1s: %d\n " "Nr. de 2s: %d\n " "Nr. de 3s: %d\n " "Nr. de 4s: %d\n " "Nr. de 5s: %d", counters[0], counters[1], counters[2], counters[3], counters[4], counters[5]);</pre>
Resultado esperado:	Que não hajam ocorrências dos números 2 e 4.
Resultado:	Teste concluído com sucesso.

Nome do teste:	Sobreposição de imagens.
Descrição:	<p>Testar se a imagem devolvida pela função implementada em 'UIImage+Combine.m':</p> <pre> +(UIImage *)imageFromImage: (UIImage *)↵ backgroundImage andOverlayedImage: (UIImage*)overlayImage; </pre> <p>retorna a imagem correspondente à sobreposição da imagem <i>backgroundImage</i> com a imagem <i>overlayImage</i>.</p>
Método:	Chamar a função "imageFromImage" com duas imagens distintas como argumentos, gravar a imagem resultante e verificar se foi realmente sobreposta.
Resultado esperado:	<p>Uma imagem correspondente à sobreposição das duas imagens.</p> <p>Código do teste:</p> <pre> UIImage *backgroundImage = [UIImage imageNamed:@"↵ image1.png"]; UIImage *overlayImage = [UIImage imageNamed:@"↵ image2.png"]; // Sobrepor "backgroundImage" com "overlayImage" UIImage *combinedImage = [UIImage imageFromImage:↵ backgroundImage andOverlayedImage:overlayImage↵]; NSData *data = UIImagePNGRepresentation(↵ combinedImage); [data writeToFile:@"combinedImage.png" atomically:↵ YES]; [data release]; </pre>
Resultado:	Teste concluído com sucesso.

Conclusões e trabalho futuro

O objectivo de criar uma aplicação para melhorar o ouvido musical foi alcançado.

Para realizar este trabalho foi necessário aprender Objective-C, uma linguagem orientada a objectos não leccionada durante o curso; XCode, um ambiente de desenvolvimento integrado da Apple; Interface Builder, um programa para criar interfaces de utilizador; e *Instruments*, um analisador de *performance* para, entre outras coisas, detecção de fugas de memória. Foi também imprescindível aprender algumas APIs do SDK iOS 4 (iPhone).

O desenvolvimento foi feito no XCode e testado numa primeira fase no simulador e numa segunda fase num aparelho físico.

Os principais obstáculos na realização deste trabalho foram: dominar o sistema de gestão de memória da linguagem de programação devido a ser pouco usual e aprender o SDK iOS 4.

Ao longo do processo de desenvolvimento iterativo foram detectados vários problemas. Por exemplo, o primeiro sistema de som implementado com a API de som do SDK iOS 4 tinha um comportamento bom no simulador mas, quando testado num dispositivo físico a *performance* era inaceitável. Os sons não eram tocados de seguida e o tempo de resposta era muito elevado. O problema foi resolvido utilizando um sistema de som *open source*, desenvolvido por Tomáš Znamenáček, que usa como API o OpenAL. Outro problema foi: como guardar os dados da aplicação? Ao início foi ponderada a ideia de usar uma base de dados SQLite mas depois de alguma investigação concluiu-se que o *overhead* gerado pela BD não era o mais indicado para aplicação em questão. Por isso, a escolha caiu sobre a serialização de objectos.

Como resultado final obteve-se uma aplicação funcional, simples de usar, e proveitosa para todas as pessoas que tocam guitarra, principalmente, iniciantes.

Como trabalho futuro, podem ser criadas aplicações com base nesta mas, com outros instrumentos, novos modos, mini-jogos, um teste de avaliação de conhecimentos para saber à partida o nível a que se encontra o utilizador, melhorar o aspecto geral da aplicação experimentando outras combinações de cor e elementos de interface. O programa pode ser igualmente convertido e adaptado a outras plataformas móveis e não-móveis, tais como, Android, Symbian, Windows, Mac OS X e Linux.

Bibliografia

- [1] Apple. iOS Dev Center. Revisto em Julho de 2010. Consultado em Junho de 2010. <http://developer.apple.com/devcenter/ios/index.action>.
- [2] iPhone SDK Articles. Localizing iPhone Apps - Internationalization. Revisto em 16 de novembro de 2008. Consultado em Agosto de 2010. <http://www.iphonesdkarticles.com/2008/11/localizing-iphone-apps.html>.
- [3] Stack Overflow. <http://stackoverflow.com/>.
- [4] Sandercoe, Justin. Practical Musical Theory. Londres: Chocolate Cake Productions Ltd, 2009.
- [5] Wei-Meng, Lee. Beginning iPhone® SDK Programming with Objective-C®. Indianapolis: Wiley Publishing, 2010.
- [6] Wikipedia. Chord (music) — Wikipedia, The Free Encyclopedia. Revisto em Setembro 2010. Consultado em Agosto de 2010. [http://en.wikipedia.org/wiki/Chord_\(music\)](http://en.wikipedia.org/wiki/Chord_(music)).
- [7] Wikipedia. Musical theory — Wikipedia, The Free Encyclopedia. Revisto em Novembro 2008. Consultado em Junho de 2010. http://en.wikipedia.org/wiki/Musical_theory.
- [8] Wikipedia. Note — Wikipedia, The Free Encyclopedia. Revisto em Setembro 2010. Consultado em Junho de 2010. <http://en.wikipedia.org/wiki/Note>.
- [9] Wikipedia. Strum — Wikipedia, The Free Encyclopedia. Revisto em Março 2010. Consultado em Junho de 2010. <http://en.wikipedia.org/wiki/Strum>.
- [10] Znamenáček, Tomáš. Boskovice. Simple, low-latency OpenAL sound engine for Apple iOS. Revisto em 4 de Setembro de 2010. Consultado em Agosto de 2010. <http://github.com/zoul/Finch>.

Anexos

Tabela 1: Solfejo em notação românica e inglesa.

Românico	Inglês
Dó	C
Dó \sharp	C \sharp
Ré	D
Ré \sharp	D \sharp
Mi	E
Fá	F
Sol	G
Sol \sharp	G \sharp
Lá	A
Lá \sharp	A \sharp
Si	B
Dó	C

Tabela 2: Designação dos intervalos entre duas notas.

Intervalo	Nome do intervalo	Número de semitons
C para C	Uníssonos	0
C para C \sharp	Uníssonos aumentado	1
C para D	2ª maior	2
C para D \sharp	2ª aumentada	3
C para E \flat	3ª menor	3
C para E	3ª maior	4
C para F	4ª justa	5
C para F \sharp	4ª aumentada	6
C para G \flat	5ª diminuta	6
C para G	5ª justa	7
C para G \sharp	5ª aumentada	8
C para A	6ª maior	9
C para A \sharp	6ª aumentada	10
C para B	7ª maior	11
C para C	8ª justa	12

Figura 1: Diagrama de classes do User.

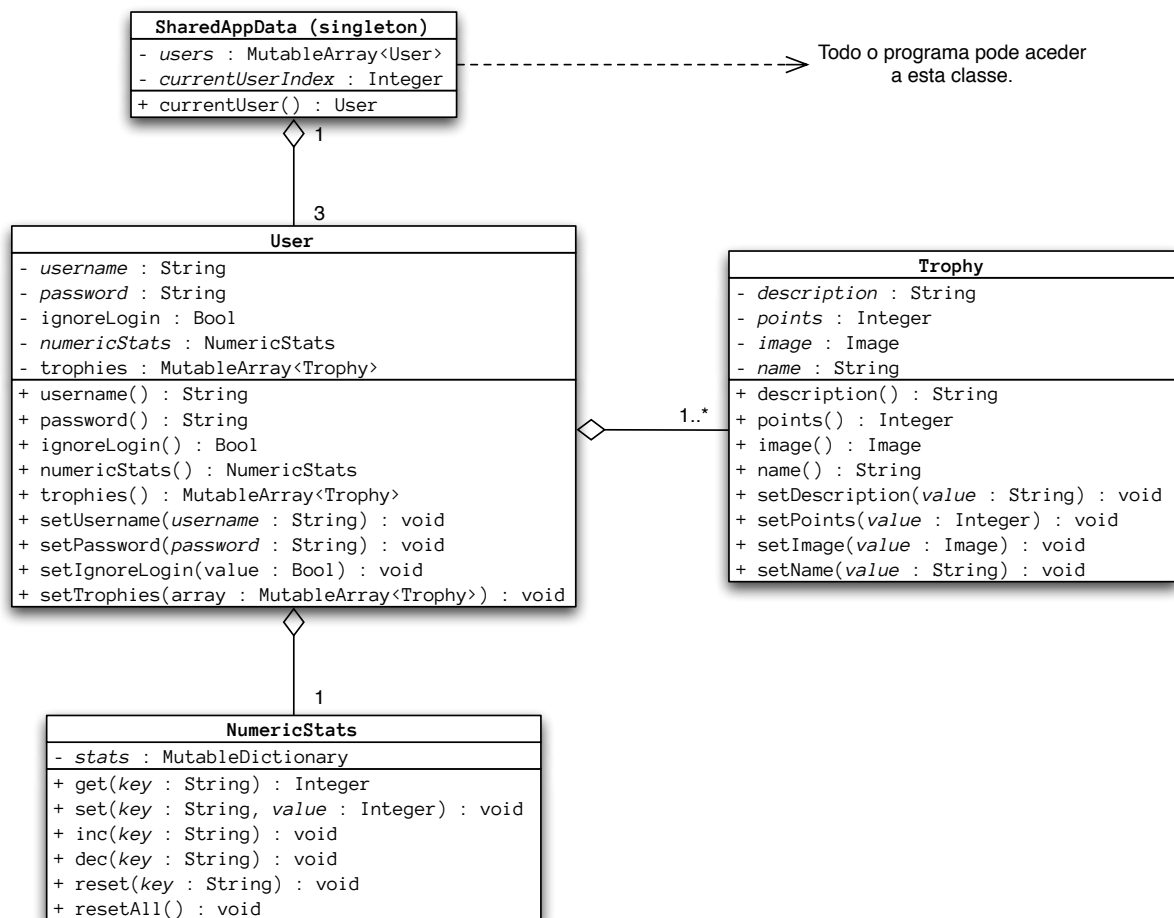


Figura 2: Diagrama de classes geral.

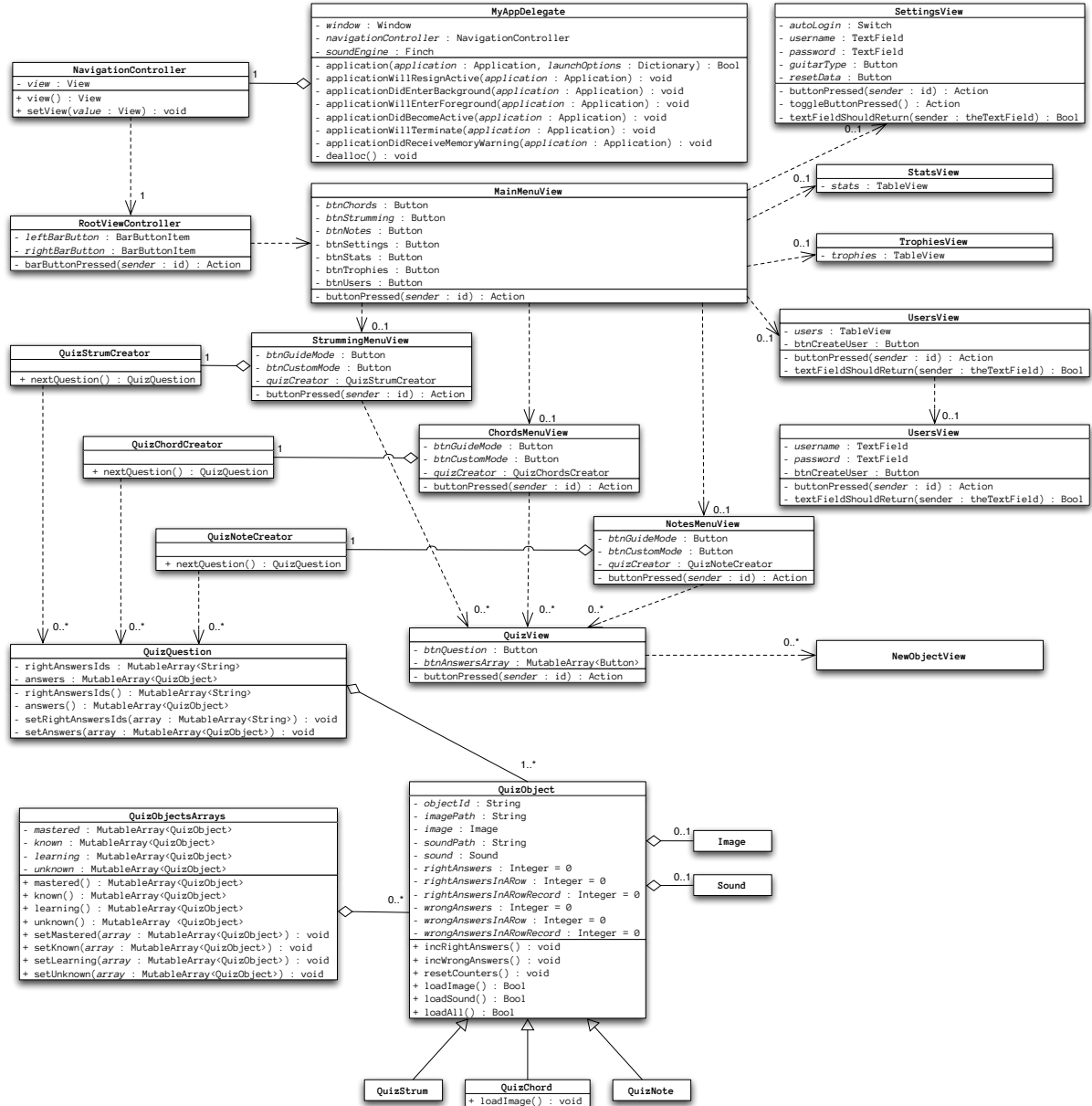


Figura 3: Diagrama de classes geral (revisão).

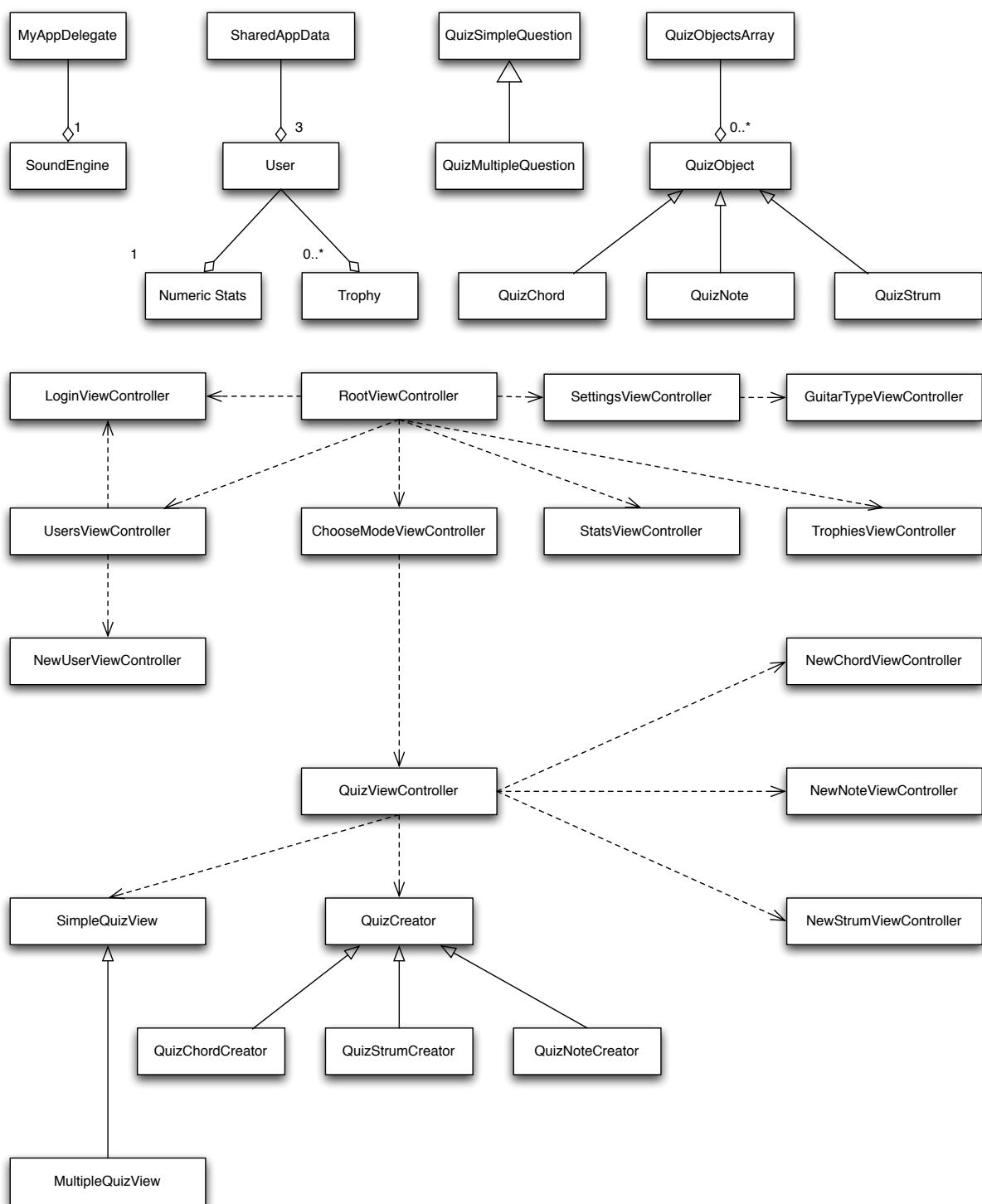


Figura 4: Fluxograma do arranque da aplicação.

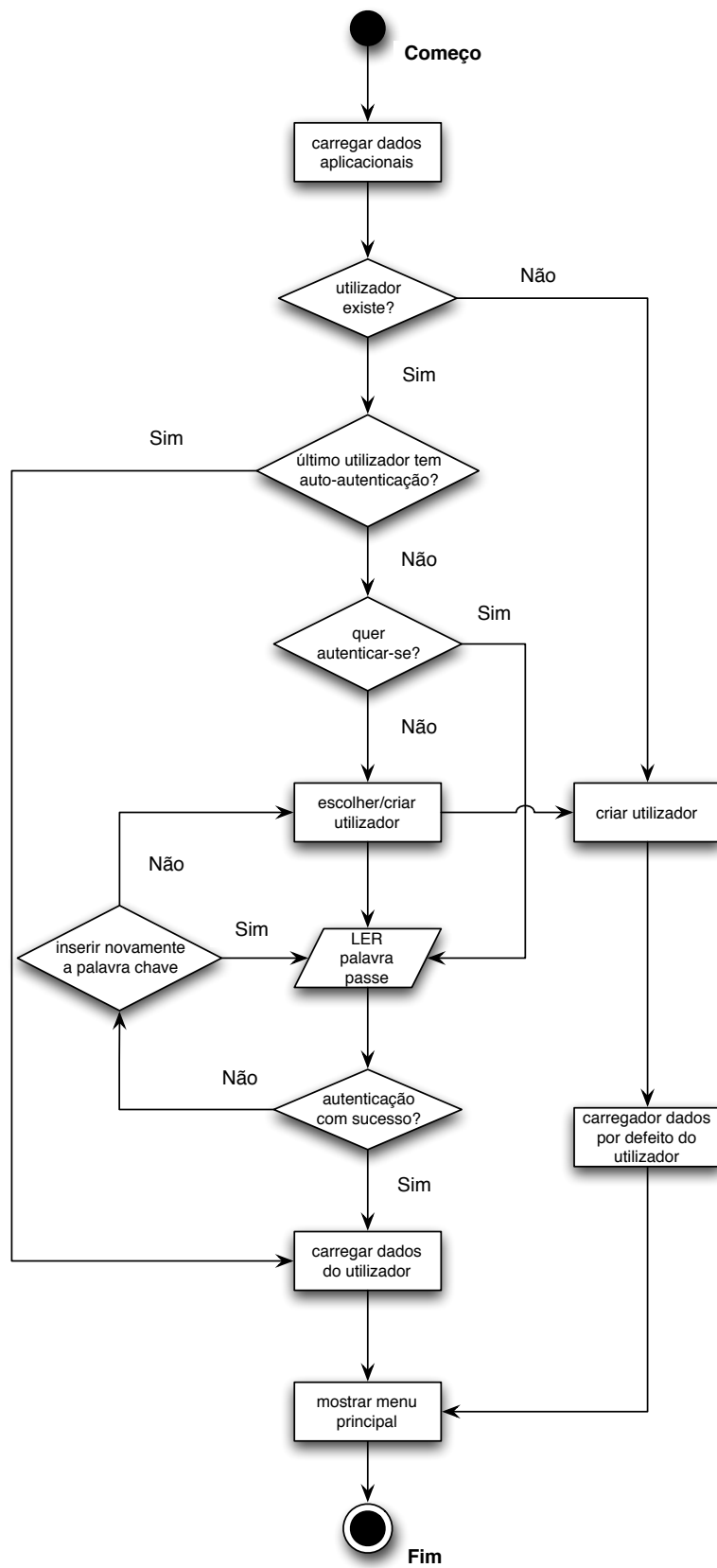


Figura 5: Mapa de navegação dos ecrãs da aplicação.

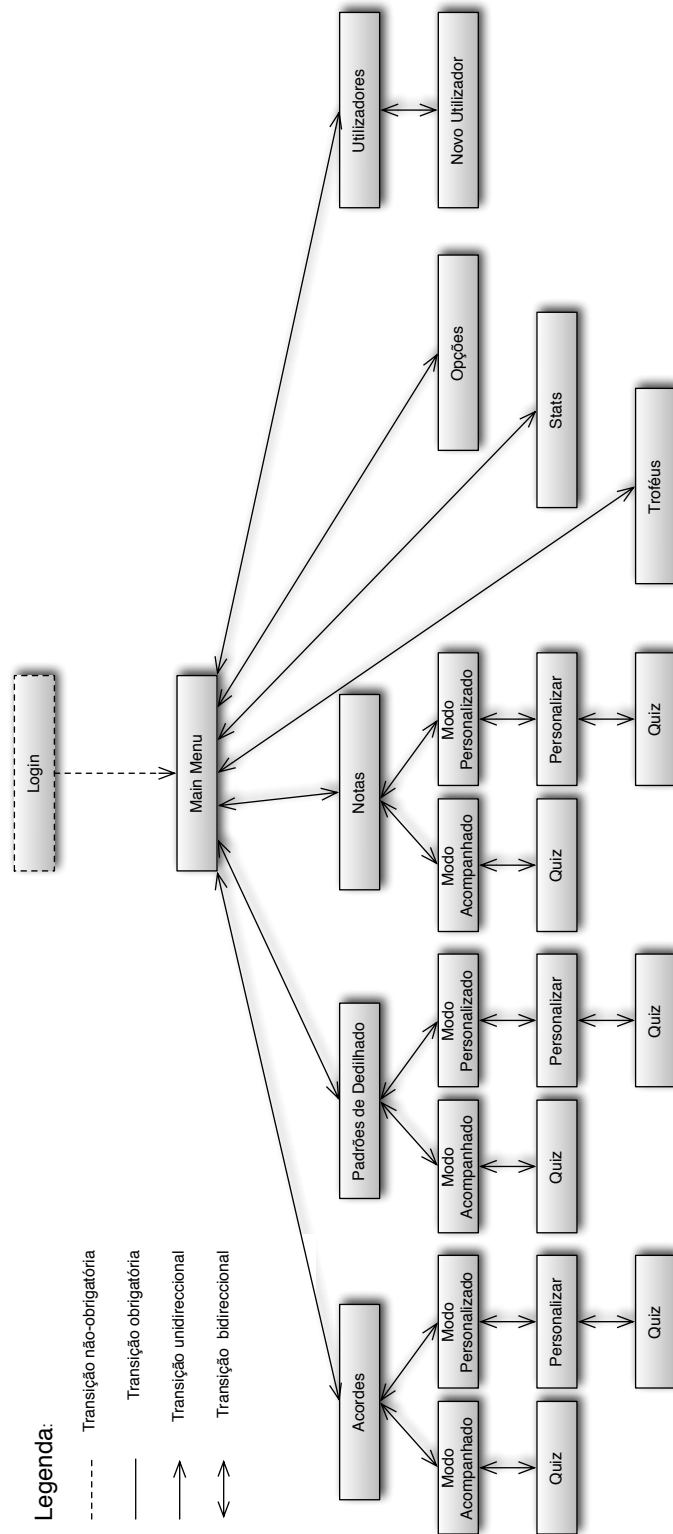


Figura 6: *Layout* dos seguintes menus: principal, acordes e opções.

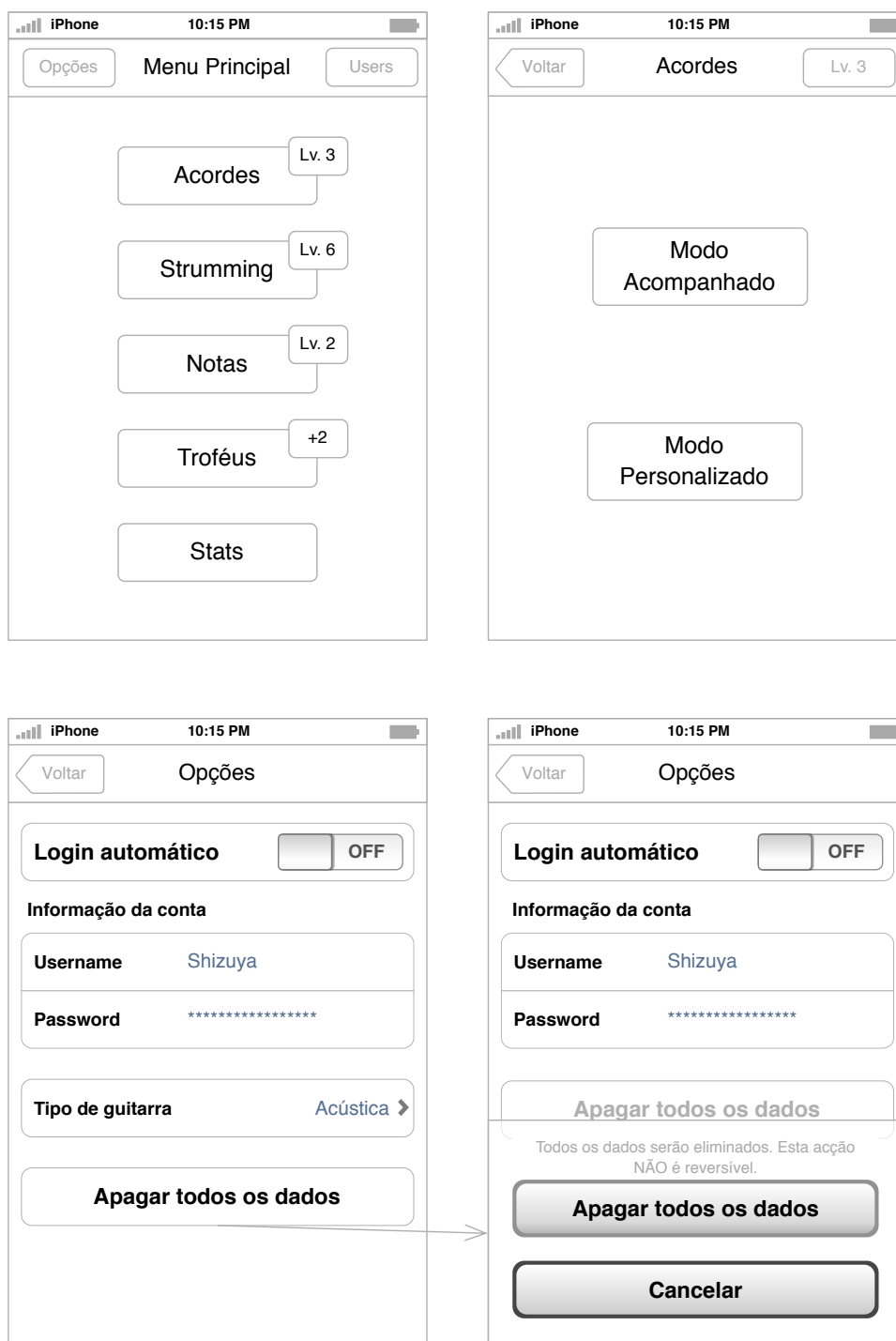


Figura 7: *Layout* dos seguintes menus: modo acompanhado e utilizadores.

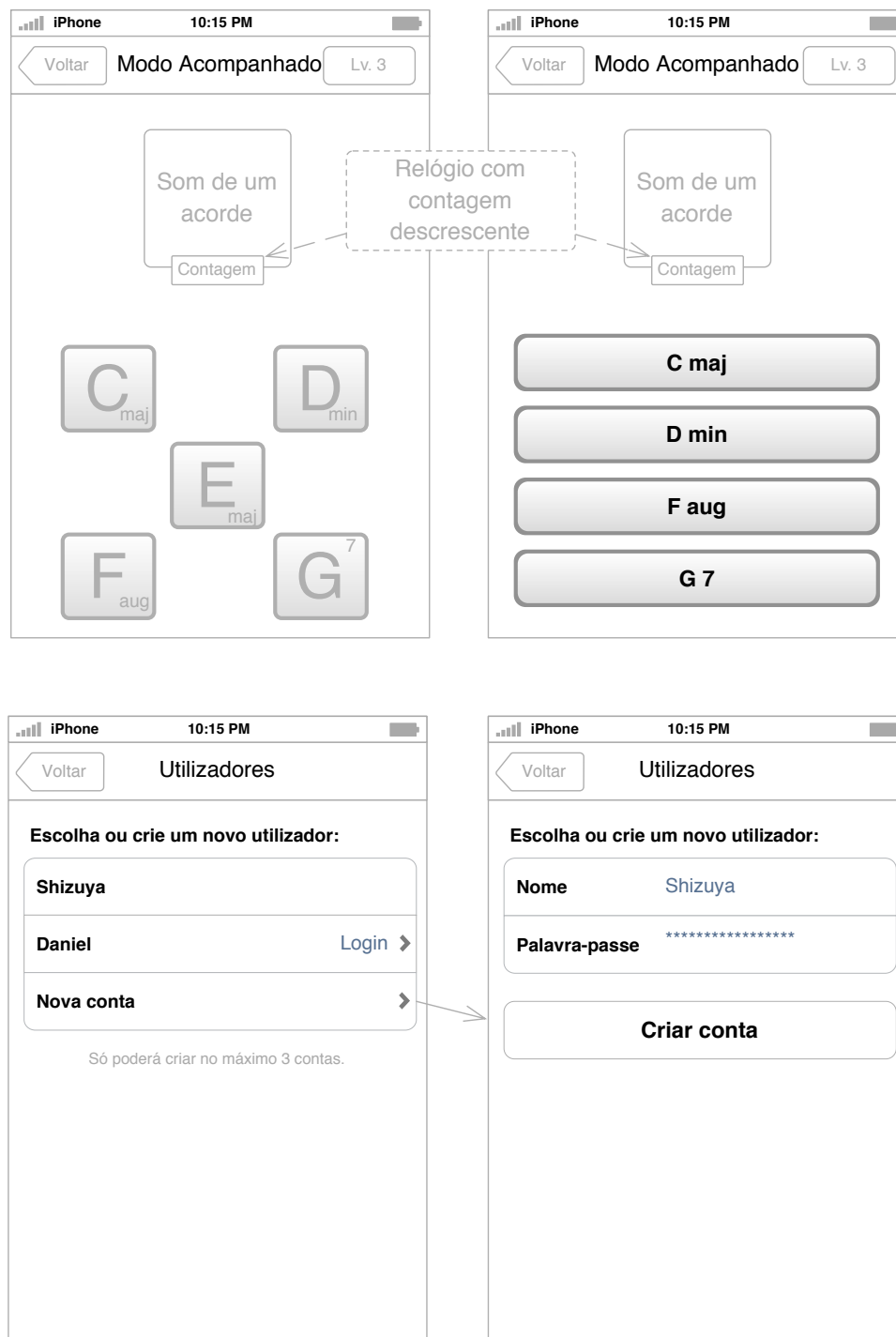


Figura 8: *Layout* dos seguintes menus: troféus e dados estatísticos.



Figura 9: Captura dos ecrãs 'Opções' e 'Opções' quando se clica em repor dados.



Figura 10: Capturas dos ecrãs 'Utilizadores' e 'Autenticação'



Figura 11: Captura do ecrã 'Acordes'.

Glossário

Arpejo	Tocar as notas de um acorde em sequência, uma atrás da outra, em vez de simultaneamente
BD	Base de Dados
Compasso	Segmento de tempo definido pelo número de batidas de uma dada duração. Um compasso 4/4 significa que há quatro batidas ou semínimas ♩ por compasso. Uma peça musical consiste em vários compassos.
Downstroke	Tocar todas as cordas de um acorde num só movimento descendente
Escala cromática	Escala musical constituída por 12 semitons
Fret	Palavra inglesa que significa ‘trasto’
Log in	Autenticar
Logged in	Estado em que se encontra um utilizador depois de se autenticar
Nota	Termo empregado para designar o elemento mínimo de um som
Nota base	Nota mais grave do acorde que dá o nome a este mesmo
Open Source	Expressão inglesa para software de código aberto
Padrão de dedilhado	Sequência repetitiva em que uma ou mais cordas da guitarra são tocadas
Power chords	Acordes sem terça
Quinta aumentada	Intervalo de oito semitons da nota base
Quinta diminuta	Intervalo de seis semitons da nota base
Quinta justa	Intervalo de sete semitons da nota base
Quiz	Conjunto de perguntas para avaliação de conhecimentos
Segunda maior	Intervalo de dois semitons da nota base
Semitom	Um semitom corresponde ao menor intervalo utilizado na escala diatónica
Semínima	Nota musical cuja duração é de 1/4
Strumming	Palavra inglesa que significa ‘dedilhar’
Strumming pattern	Palavra inglesa que significa ‘padrão de dedilhado’
Terça maior	Intervalo de quatro semitons da nota base
Terça menor	Intervalo de três semitons da nota base
Trasto	Os filetes de metal que atravessam o ponto dos instrumentos de cordas dedilhadas, para indicar onde o tocador deve pôr os dedos
Triade	Acorde formado por três notas de uma escala
Trigger	Um evento que causa outro(s) evento(s)
Upstroke	Tocar, normalmente, as três cordas mais agudas de um acorde num só movimento ascendente
Acorde	Grupo de notas tocadas em simultâneo
IDE	Integrated Development Environment (Ambiente Integrado de Desenvolvimento)