

Tagarela: Aplicativo para Comunicação Alternativa no iOS

Alan Filipe Cardozo Fabeni
prof. Dalton Solano dos Reis

FURB - Universidade Regional de Blumenau
DSC - Departamento de Sistemas e Computação
Grupo de Pesquisa em Computação Gráfica e Entretenimento Digital
2012/II - <http://www.inf.furb.br/gcg/>



ROTEIRO

- INTRODUÇÃO
- FUNDAMENTAÇÃO TEÓRICA
- DESENVOLVIMENTO
- IMPLEMENTAÇÃO
- CONCLUSÃO
- DEMONSTRAÇÃO

INTRODUÇÃO

CONTEXTUALIZAÇÃO

Inclusão social através da comunicação



Comunicação alternativa



Criação de uma ferramenta genérica



Recursos multimídia e troca de mensagens entre os usuários envolvidos

INTRODUÇÃO

OBJETIVOS

- Desenvolver um aplicativo para comunicação alternativa utilizando os recursos multimídia da plataforma iOS
- Especificar um ambiente que possa ser customizado de forma a atender as necessidades específicas de cada paciente através da troca de mensagens entre as pessoas envolvidas

FUNDAMENTAÇÃO TEÓRICA

TECNOLOGIA ASSISTIVA

- Reúne recursos e serviços que proporcionam habilidades para indivíduos com necessidades especiais
- Recursos: equipamentos utilizados para melhorar as capacidades do indivíduo
- Serviços: são prestados de forma profissional, visando o uso dos recursos

FUNDAMENTAÇÃO TEÓRICA

COMUNICAÇÃO ALTERNATIVA

- Área da tecnologia assistiva que auxilia os indivíduos com necessidades especiais a se comunicarem
- Atende indivíduos sem fala ou escrita funcional

FUNDAMENTAÇÃO TEÓRICA

PRANCHAS DE COMUNICAÇÃO

- Objetos utilizados para transmitir mensagens através de símbolos gráficos
- Os símbolos representam algum objeto ou conceito do mundo real
- O usuário forma sentenças visuais com estes símbolos

FUNDAMENTAÇÃO TEÓRICA

TIPOS DE SÍMBOLOS



Blissymbolics



Pictogram Ideogram Communications



Pictogram Communication System

FUNDAMENTAÇÃO TEÓRICA

PLATAFORMA IOS - IOS

■ Sistema operacional móvel da Apple



Cocoa Touch

Media

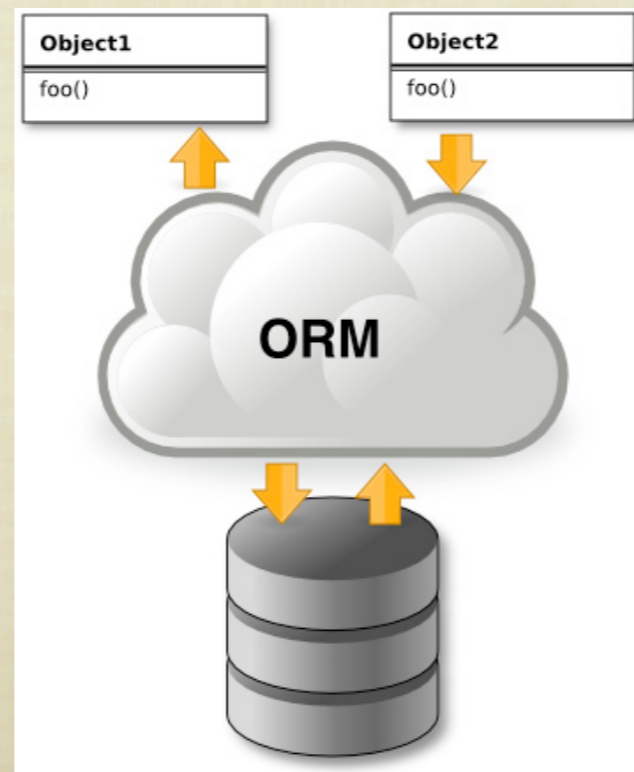
Core Services

Core OS

FUNDAMENTAÇÃO TEÓRICA

PLATAFORMA IOS - CORE DATA

- Framework da Apple que implementa um mapeamento de objeto relacional
- Otimizado para dispositivos móveis
- Abstrai os comandos SQL para um paradigma de orientação a objetos



FUNDAMENTAÇÃO TEÓRICA

PLATAFORMA IOS - ICLOUD

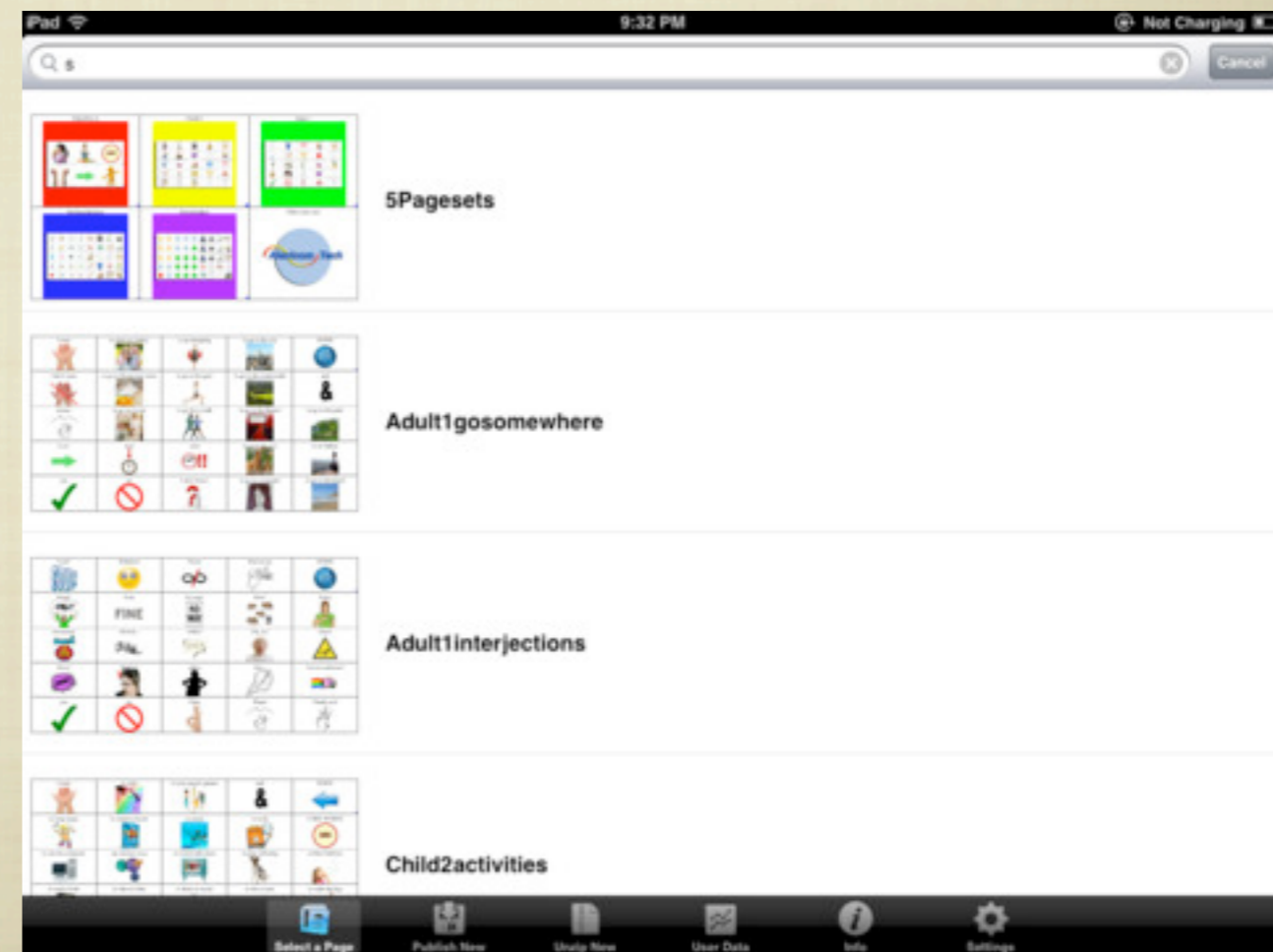
- Serviço de armazenamento em nuvem da Apple



FUNDAMENTAÇÃO TEÓRICA

TRABALHOS CORRELATOS




ALEXICOM AAC



FUNDAMENTAÇÃO TEÓRICA

TRABALHOS CORRELATOS

PROJETO AMPLISOFT

 eu	 comer				
 Limpar	 Voltar	 Falar	 Opções	 Perfil	 Fechar
 Expre. Sociais	 Pessoas	 Substantivos	 Verbos	 Adjetivos	 Miscelânea
 melancia	 banana	 amendoim			

DESENVOLVIMENTO

ESPECIFICAÇÃO - REQUISITOS FUNCIONAIS

O aplicativo deverá permitir que...

- o fonoaudiólogo crie pacientes
- o fonoaudiólogo crie tutores
- o fonoaudiólogo crie categorias de símbolos
- o fonoaudiólogo crie símbolos
- o fonoaudiólogo crie planos de atividades
- o fonoaudiólogo envie os planos criados
- o tutor do paciente faça a importação dos planos criados pelo fonoaudiólogo
- o paciente interaja com os símbolos através do toque

DESENVOLVIMENTO

ESPECIFICAÇÃO - REQUISITOS NÃO FUNCIONAIS

O aplicativo deverá...

- rodar no sistema operacional iOS 5 e em suas versões superiores
- ser implementado utilizando a linguagem de programação Objective-C
- ser implementado utilizando o ambiente de desenvolvimento Xcode
- seguir os guias de interface visual da Apple
- realizar a troca de informações entre o fonoaudiólogo e o tutor via arquivos no formato JSON

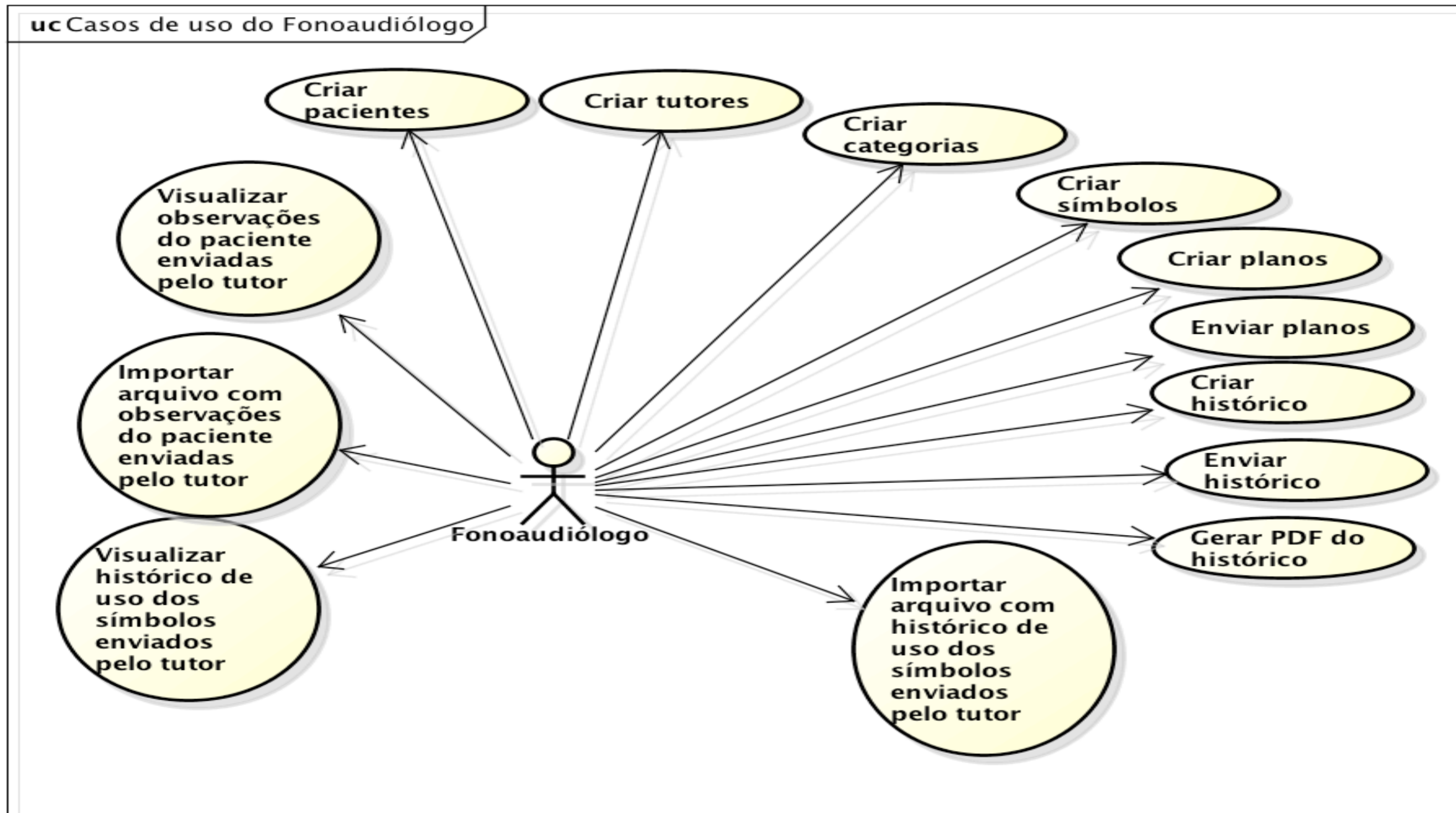
DESENVOLVIMENTO

ESPECIFICAÇÃO - TÉCNICAS E FERRAMENTAS UTILIZADAS

- Técnicas utilizadas -> UML e MER
- Ferramentas utilizadas -> Astah Professional e Xcode

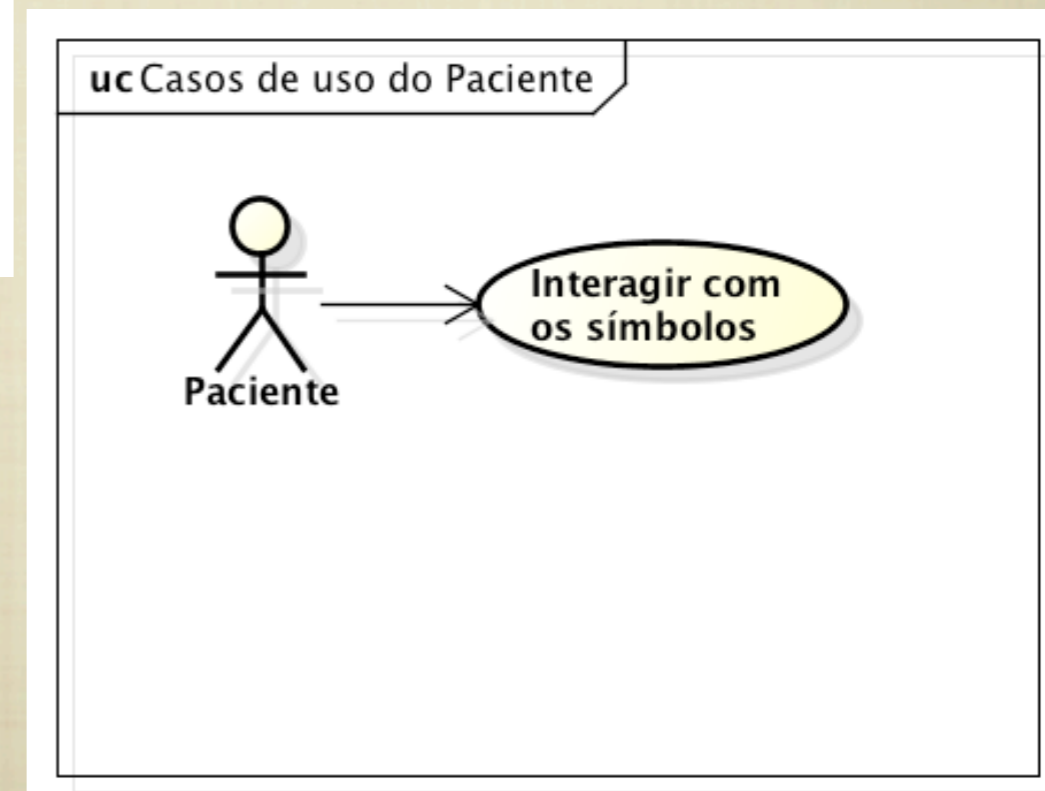
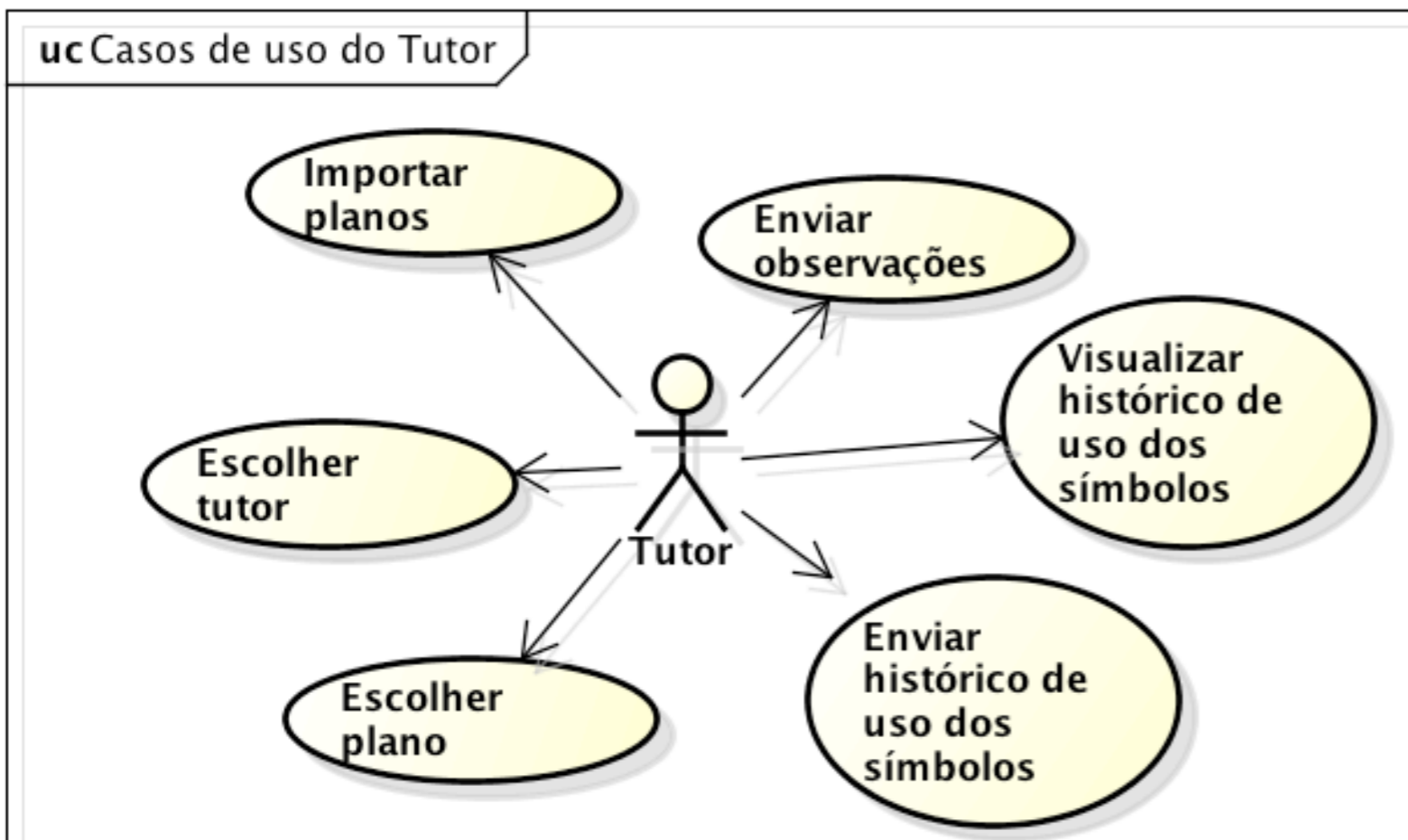
DESENVOLVIMENTO

ESPECIFICAÇÃO - CASOS DE USO



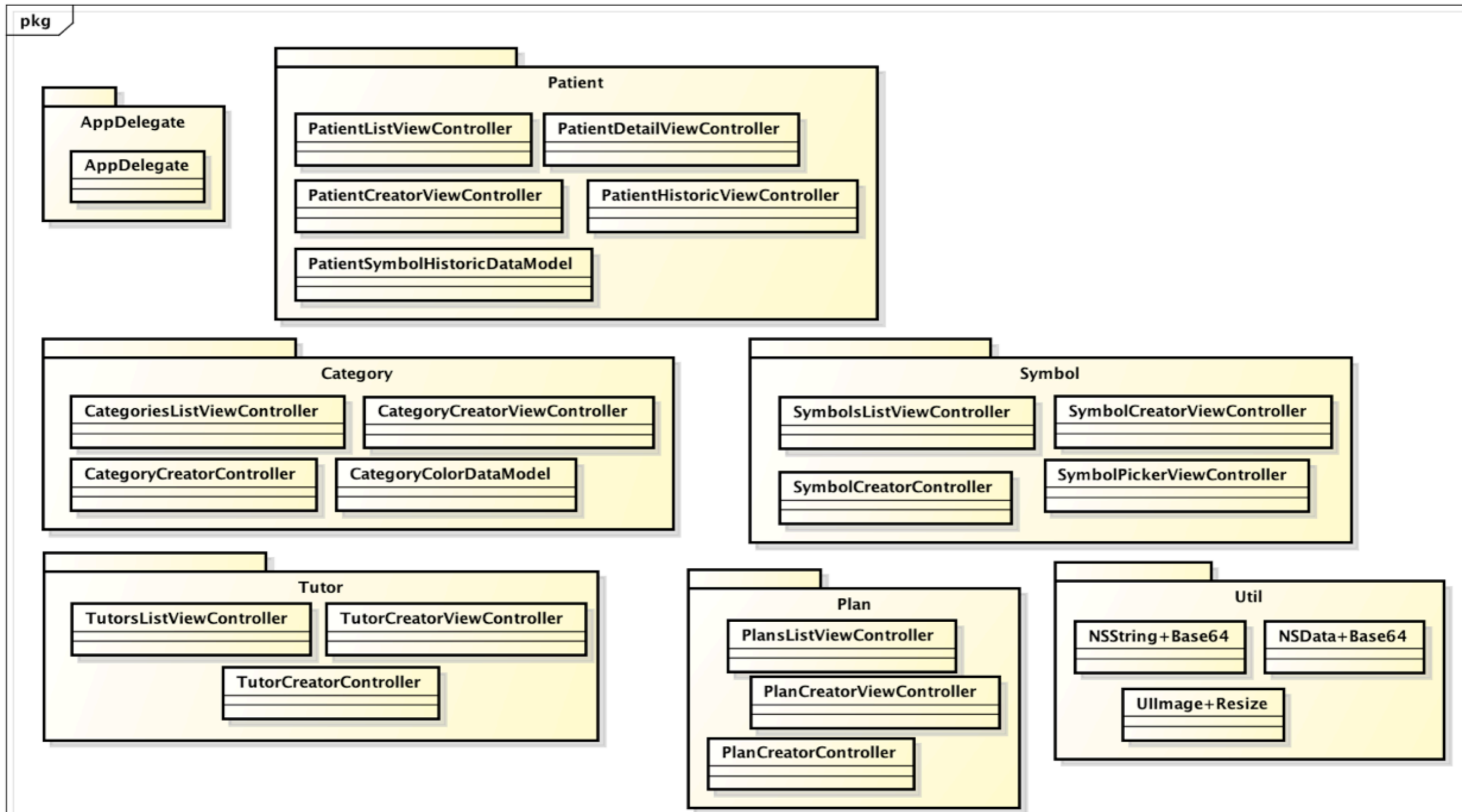
DESENVOLVIMENTO

ESPECIFICAÇÃO - CASOS DE USO



DESENVOLVIMENTO

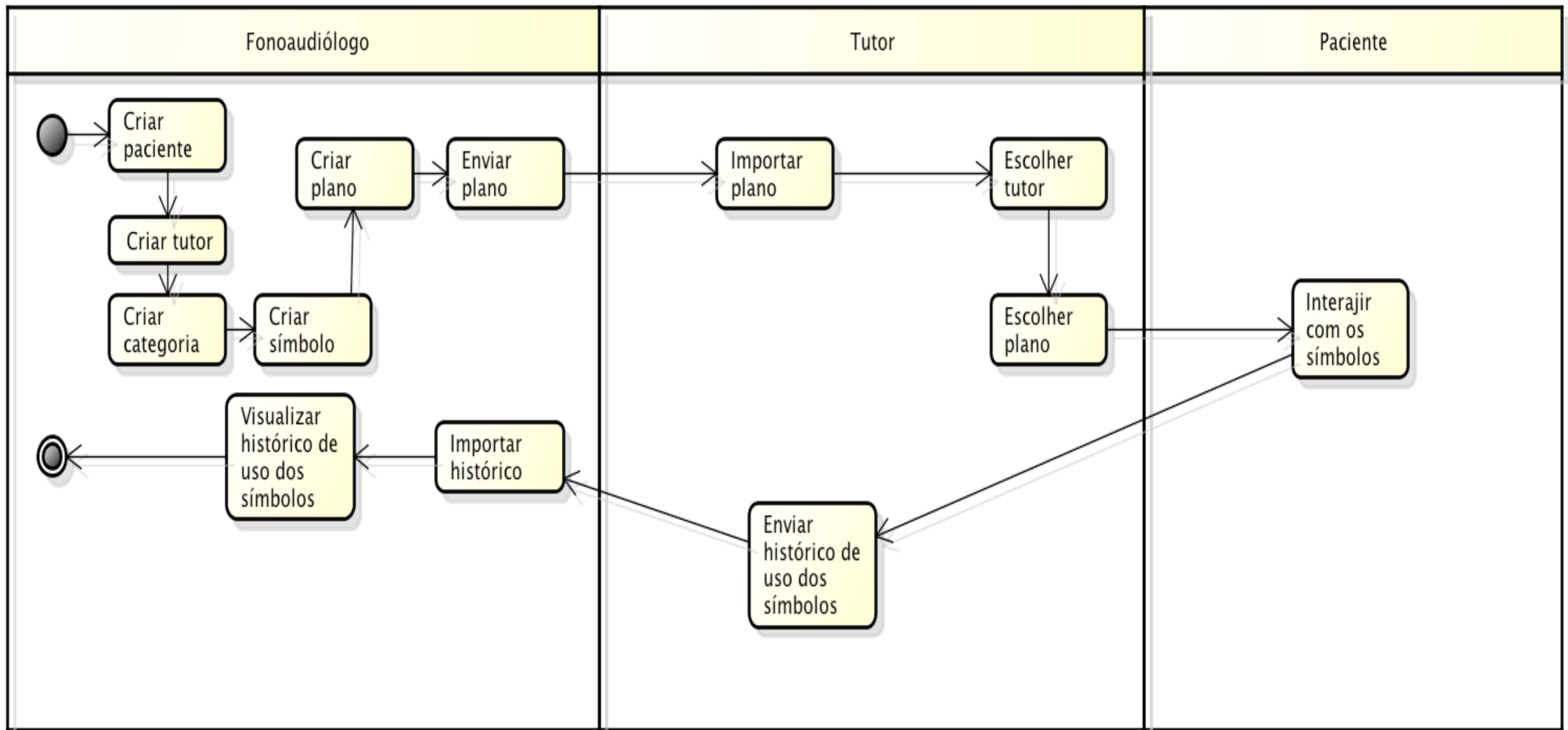
ESPECIFICAÇÃO - DIAGRAMA DE PACOTES



DESENVOLVIMENTO

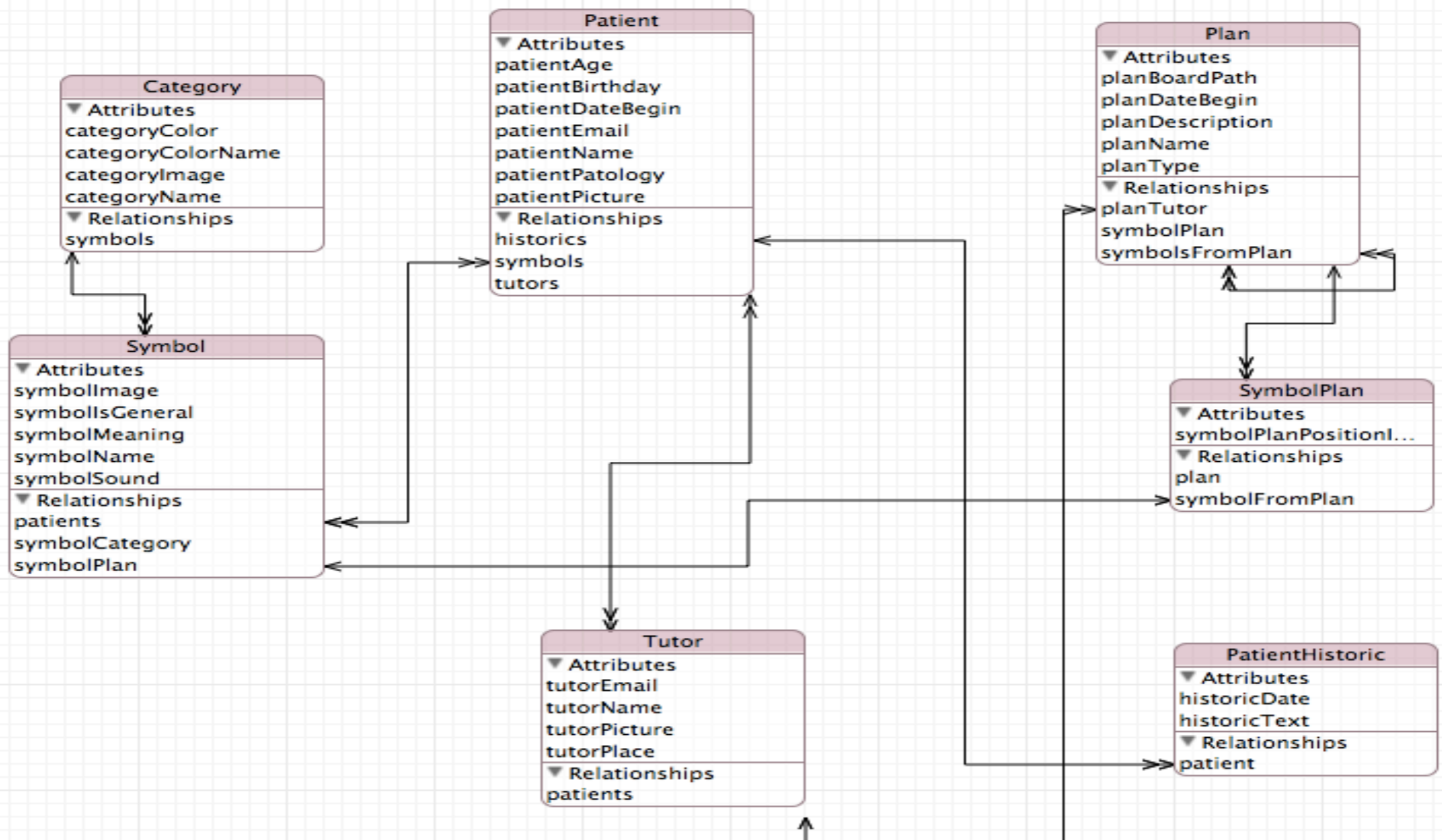
ESPECIFICAÇÃO - DIAGRAMA DE ATIVIDADES

act Uso completo do protótipo



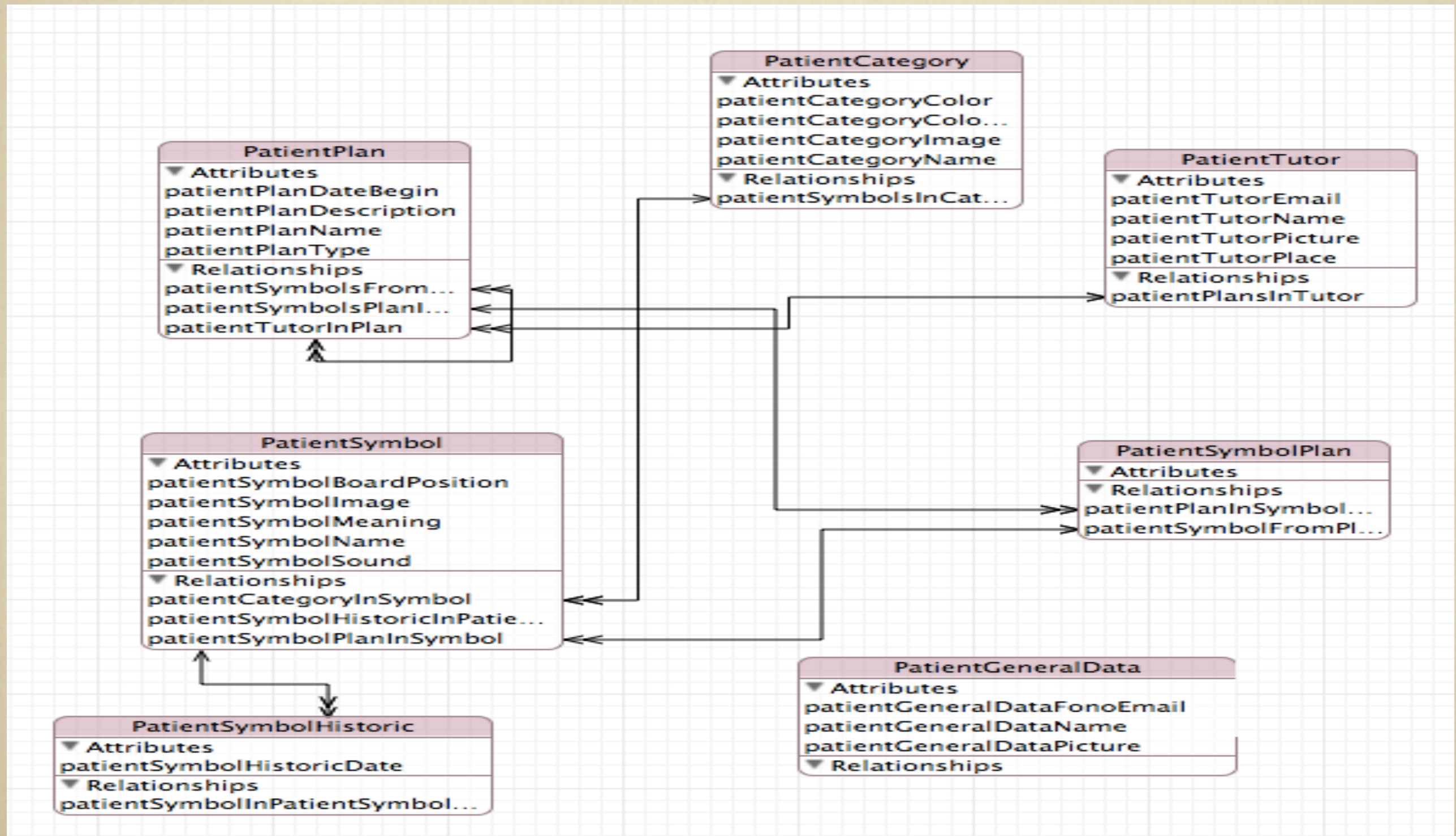
DESENVOLVIMENTO

ESPECIFICAÇÃO - MODELO DE ENTIDADES E RELACIONAMENTOS



DESENVOLVIMENTO

ESPECIFICAÇÃO - MODELO DE ENTIDADES E RELACIONAMENTOS

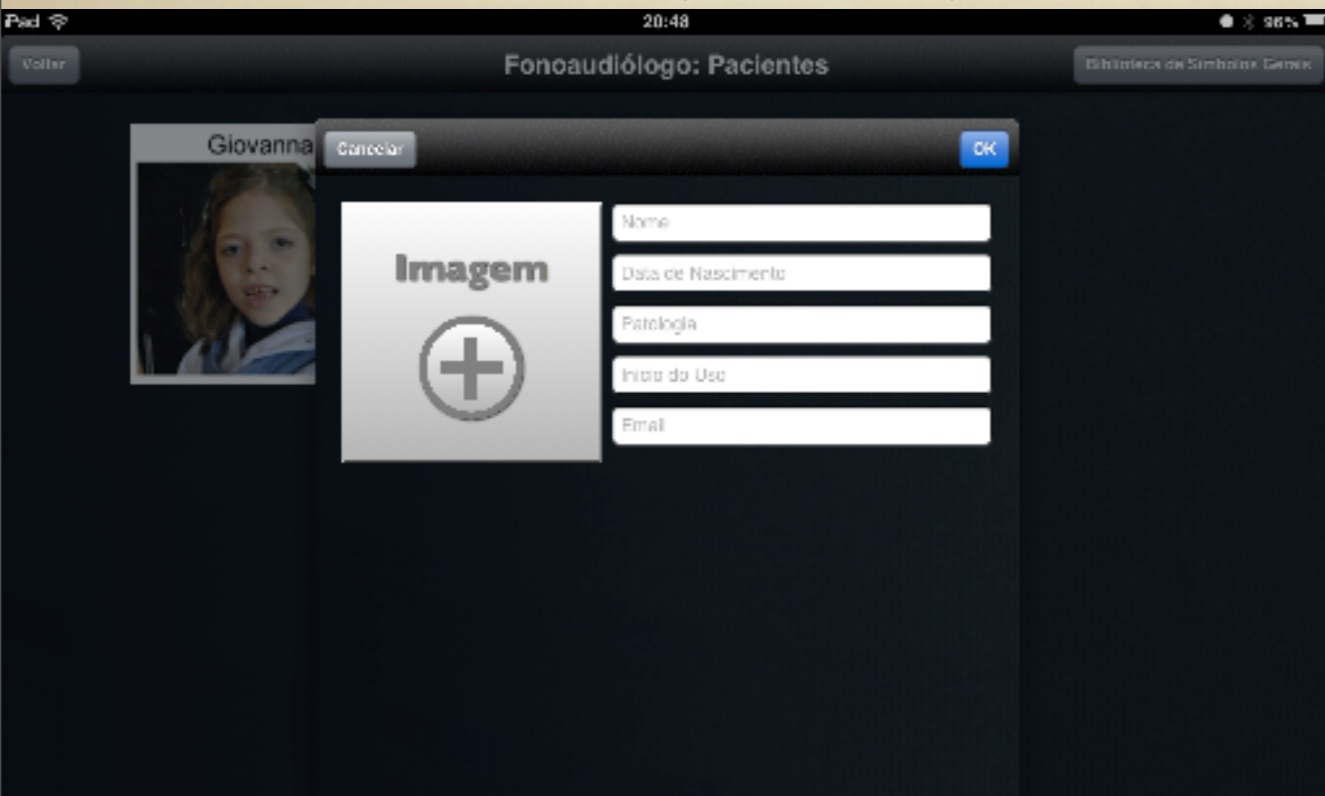


IMPLEMENTAÇÃO

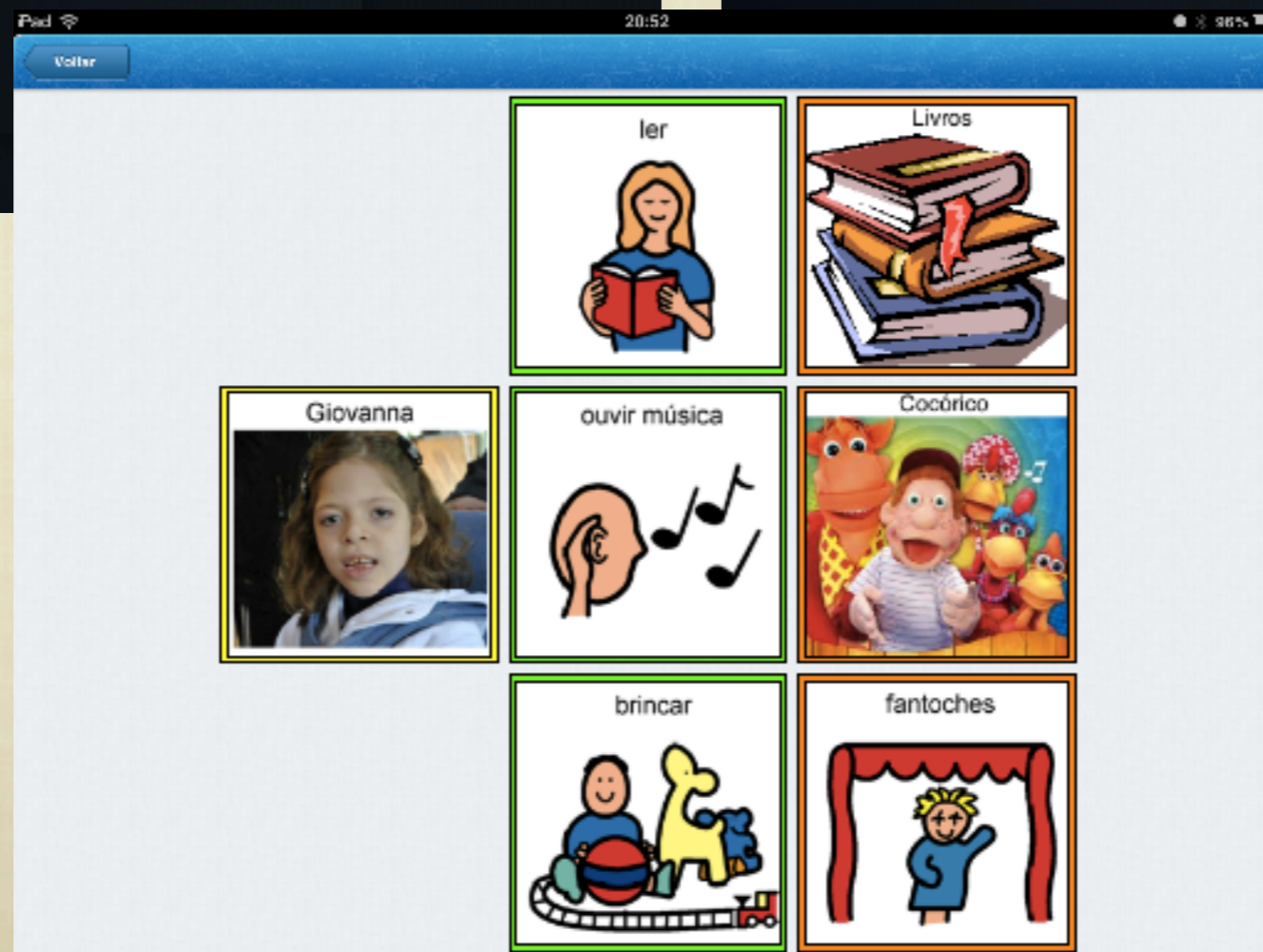
TÉCNICAS E FERRAMENTAS UTILIZADAS

- Técnicas utilizadas -> Orientação a objetos e MVC (Model View Controller)
- Ferramentas utilizadas -> iOS SDK, Xcode, Instruments e iOS Simulator
- 3 iPads de segunda geração para os testes com o aplicativo

IMPLEMENTAÇÃO



OPERACIONALIDADE



IMPLEMENTAÇÃO

CRIAÇÃO DO ARQUIVO JSON

```
87 - (void)createJSONFileWithPlan:(Plan*)plan andPatient:(Patient*)patient
88 {
89     NSMutableArray *arrayOfJSONData = [[NSMutableArray alloc] initWithCapacity:0];
90     NSError *error;
91     //1. O método recebe como parâmetro os dados do plano que será enviado e os dados do paciente para qual será enviado o plano
92     NSDictionary *infoPlan = [NSDictionary dictionaryWithObjectsAndKeys:
93         [plan planDescription], @"plan.description",
94         [dateFormat stringFromDate:[plan planDateBegin]], @"plan.datebegin",
95         [plan planType], @"plan.type",
96         [plan planName], @"plan.name",
97         [[plan planTutor]tutorName], @"tutor.name",
98         [[plan planTutor]tutorEmail], @"tutor.email",
99         [[plan planTutor]tutorPlace], @"tutor.place",
100        [NSString encodeBase64WithData:[plan planTutor]tutorPicture], @"tutor.picture",
101        [patient patientName], @"patient.name",
102        @"rodrigomarcellinofranca@hotmail.com", @"fono.email",
103        [NSString encodeBase64WithData:[patient patientPicture]], @"patient.picture",
104        nil];
105     //2. Esses dados são convertidos para o formato JSON
106     NSData* jsonDataPlan = [NSJSONSerialization dataWithJSONObject:infoPlan options:NSJSONWritingPrettyPrinted error:nil];
107
108     [arrayOfJSONData addObject:jsonDataPlan];
109
```

```
109
110     NSArray *arraySymbolsFromPlan;
111     NSFetchRequest *fetchRequest = [[NSFetchRequest alloc] init];
112     NSEntityDescription *entity = [NSEntityDescription entityForName:@"SymbolPlan" inManagedObjectContext:managedObjectContext];
113     NSPredicate *predicate = [NSPredicate predicateWithFormat:@"(ANY plan == %@)", plan];
114     //3. Busco todos os símbolos do plano recebido como parâmetro no Core Data
115     [fetchRequest setPredicate:predicate];
116     [fetchRequest setEntity:entity];
117
118     arraySymbolsFromPlan = [managedObjectContext executeFetchRequest:fetchRequest error:&error];
119
```

```

120 for (int i = 0; i < [arraySymbolsFromPlan count]; i++) {
121     NSString *planSymbolName = [[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolName];
122     NSString *planSymbolMeaning = [[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolMeaning];
123     NSString *planSymbolSound = [[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolSound];
124     NSString *planSymbolSoundData = [NSString encodeBase64WithData:[NSData dataWithContentsOfFile:planSymbolSound]];
125
126     NSString *planCategoryName = [[[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolCategory]categoryName];
127     NSString *planCategoryColorName = [[[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolCategory]categoryColorName];
128     NSString *planCategoryColor = [[[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolCategory]categoryColor];
129     NSString *planSymbolImage = [NSString encodeBase64WithData:[[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]symbolImage
130         ]];
131     NSString *planSymbolPosition = [NSString stringWithFormat:@"%@", [[arraySymbolsFromPlan objectAtIndex:i]
132         symbolPlanPositionInBoard]];
133
134     if ([planSymbolSoundData length] == 0) {
135         planSymbolSoundData = @"NULL";
136     }
137     //4. Transformo eles em formato JSON
138     NSDictionary *infoSymbolCategory = [NSDictionary dictionaryWithObjectsAndKeys:
139         planSymbolName, @"symbol.name",
140         planSymbolMeaning, @"symbol.meaning",
141         planSymbolImage, @"symbol.image",
142         planSymbolSoundData, @"symbol.sound",
143         planSymbolPosition, @"symbol.position",
144         planCategoryName, @"category.name",
145         planCategoryColorName, @"category.colorname",
146         planCategoryColor, @"category.color",
147         [NSString encodeBase64WithData:[[[arraySymbolsFromPlan objectAtIndex:i]symbolFromPlan]
148             symbolCategory]categoryImage]], @"category.image", nil];
149     NSData* jsonDataSymbolCategory = [NSJSONSerialization dataWithJSONObject:infoSymbolCategory options:
150         NSJSONWritingPrettyPrinted error:&error];
151     [arrayOfJSONData addObject:jsonDataSymbolCategory];
152 }

```

```

150 NSMutableString *textToWrite = [[NSMutableString alloc] initWithCapacity:0];
151 fileName = [NSString stringWithFormat:@"%s/%s_plan_for_%s_%.txt", DOCUMENTS_FOLDER, [[plan planTutor]tutorName], [patient
152     patientName], [plan planDateBegin]];
153 //5. Faça a concatenação entre esses dois JSON gerados
154 for (int i = 0; i < [arrayOfJSONData count]; i++) {
155     NSData *jsonData = [arrayOfJSONData objectAtIndex:i];
156     NSString *contents = [[NSString alloc] initWithData:jsonData encoding:NSUTF8StringEncoding];
157     [textToWrite appendString:[NSString stringWithFormat:@"%s$", contents]];
158 }
159 //6. Crio um arquivo .txt com essa concatenação
160 [textToWrite writeToFile:fileName atomically:NO encoding:NSUTF8StringEncoding error:&error];
161 [self insertCreatedPlan];
162
163 if (error != nil) {
164     NSLog(@"Erro ao salvar o arquivo: %@", error);
165 }

```

IMPLEMENTAÇÃO

CRIAÇÃO DO ARQUIVO JSON

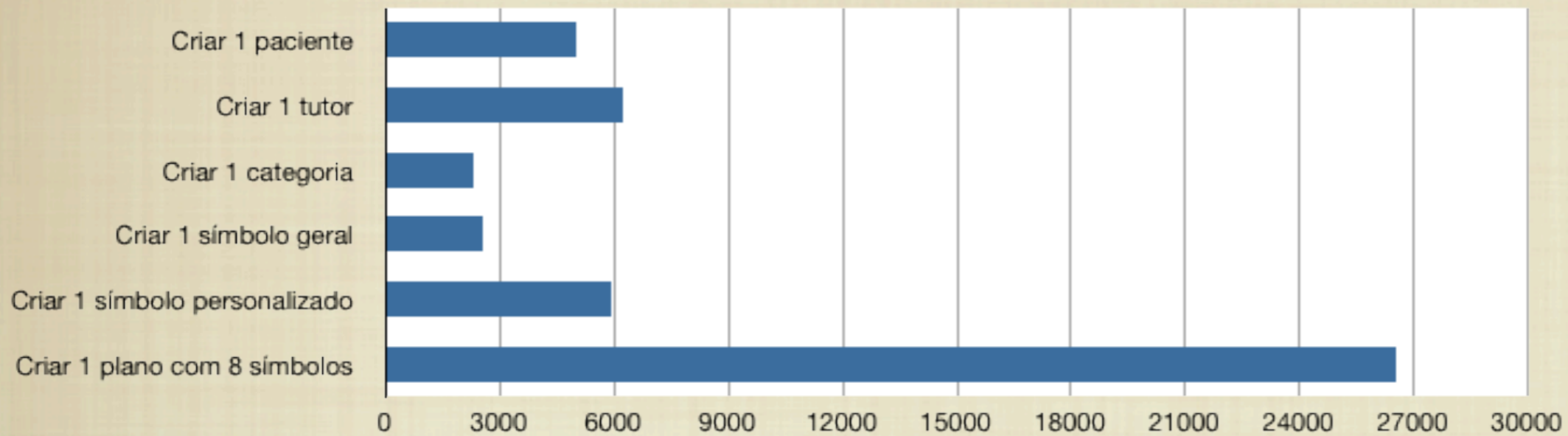
Trecho do arquivo JSON transmitido via e-mail

```
"category.colormame": "Amarelo",
"symbol.meaning": "Pai",
"category.color": "1 1 0 1",
"symbol.name": "Pai",
"category.image": "iVBORw0KGgoAAAANSUhEUgAAAAEAAAABCAyAAAIcSjAAADUIEQVQIHWP4V5VhPwAHVQL+5wKavwAAAABJRU5Erkkgg==",
"symbol.position": "3"
}$ {
"symbol.image": "iVBORw0KGgoAAAANSUhEUgAAAL4AAAC
+CAYAAACldLWdAAAAHGIET1QAAAACAAAAAAAAAAF8AAAAoAAAAAXwAAAF8AABcokAaCmAAAFvRjREFUeAHsnQm8VdMXx5s18hBKKQ2qFylNUqioNHwMyUwikkgiZCz9ZSyFBj6VniQilBJKGVYpxLJ+AihopJ52vW7Xf3X
+e9z373v3Xvfe8OZ6VP571z7z3n7LPP3r+19lpr712GePitUAAW6BMANVZvbJrAeOA70AQyBZwwA9kt7uXdsB3GAhCzjgB7Lb3Us74DsMBLIFHPAD2e3upR3wHQYC2QIO+IHsdvfSDvgOA4FsAQf8QHa7e2kHfleBQLAA77V7XV+
+af5+eefzY4dO8yvW5qW77b+vszo9c88svv8g1XPvXX38VuObfv81vW+u1xDWbV99pv5559WOU4hzVK12dxj1JR9V7xxx96qeVl79RPKVl97OdwHddTD+6lLjw7COSAH+plwLBP0yZzzz33mCOPPNlcfvjhZtCgQWbZsmU
+YAP4mTNnmuOOO8507NjRnHDCCebpp5V2gQWAVrpp+b666+Xcihr+PDh5v333xfwAVlFixebxx57zCxYsECuGTJkiNmwYYOcz8VPN2PGjDGHXaY6dSpkxkxYoR59913vXuffv55M2nSJPPTTzV58Ll9+3Zz9913myeeeEJ+B8wffV
yxueaaaOyHDh3kOVdfbVZt26dx4SAfcaMGWbVqIxmjjvukGtuwvImYXpf4Vn4JfDABVSArlevXqZatWqmZcuWarrdtdvN1KxZ0yxatEqYA5BMnDJRVKxY0bRo0UKAv99++5kqVaqYO+
+8U6BBWewXr5fz1atXN4cccoiUVbVqVdOgQQOzZs0aAdV5550nZXNV69atTdu2bc17771n1q5dK8VdtdTbt27QSwNWrUME2bNlVvV22SOc5c+YYvn7uuec8OALyZ5991lSuXNmMHZ9eqP3aa6V
JfZQF8CmPslq1aiXq52aY56CDDlKNGzc2e+21l7xTl27dCjCV96As+hB44KOGDB061AD0hx9+WICJ1AZoJ510kaIXr5756auvBAy5ubnmiCOOEJWAa5DiAOfo048Wqc+l0L9V
fykrLy9P1AdUienTpwwozj33XFF7eB4gVf322wXM27ZlM1u2bDE9evQw9evXl5EGMPO3fPlyASQjAOXDpLVq1TIXXHCBCRY5B0uwvhisVuuu4uUV+GHH8yhvx5qQVLL78s1zHSwCwAH3CragPTwRCcg3FW
PFHr9wswnmBVwk88AFJs2bNzBlInnCH6PXq76vll4JvcHFEHkI5l+ubNm5tXXnlFwA8QkeJla0D62Wefmb333lvAb7c0lLvrvmAw88llvFarPHHnuYV
JBaAwG4Tz75RIA7duxYXz1gioULF8po9OKLL8r9AwYMEPDDkNB3331nGjVqZLp37y6q2erVqW2jzOTJk31lbd261cyePVuAzjXo9lxKgN+2DaTQLP8XaOAjtQEA0hfd+nSxfr3Lmz6NioMkhowltuzfdKISqJ
+iBs2DCzYsUKkCAH12Z87NmzYoKG4xmq19qARAhPH8+INNhQoVRPUJrwdqCurNvffeK9c+VvjjAmyew3NVzZk6daoAGAAjLKR+eFkAHQKvd+5cAT4jwDHHHBO1vtl6lvDAR6UBJOigAMz+a9++venatatITgAA+O+77z7R2wEPllpTjV
9dMPI8dJLL8lv6OHhhFSH0Wzgcw8E8OfNm2fKlSsn6oldBz4DYNQTDGklDwfffc1ffv2FaaDMffcc09Rc5DcGKyUhRoWqSxGhpUrV3rA53vQKPDAR1VBLRg1alSBvgesuPcALJ9RdwAVf+j306ZNEV0e8DMA4H2hrAKTJvjK414vV
zSYHCiXqjEt4HV1FNPyb3YGeGEVKcelAPBPKg7GN+oXU2aNBElh3oCfNQZDHW8R+EEk2IZ1AWJ74Af3koB+l66gVQEAAARkAEwwP3OO++laVGDDz4QV
R1XJ54dwMM1XlMLEDsAtQfDEIOS63AvKsPwjIEDBxq8OF988YUyoqg6CnzKyGvp+7Vr1xaDGmDrvXxeunSpufXWW8Xlpt4Pm5LVK6TTz5ZRp0pU6Zlb1EWTEmdKbOlfA7ZaFSjRs3TuwCGNkBPwAgjVSKqhggQ
+OXR09nFEAHBsRl1ddff1AfeCBB4qXBx88hu+bb75pzjrrLDFKARTMgrSnLxIPCowD6MJatFpp53mqTqUq8CnXkhhVOzc03jwYPPGG2+MYFez7UwDSqO0tdff23qhzxAnNtnn32EAfUcALVoooukrMsv1w8VlxGMC0qGu5R5i2cxNcWC
+gRwDKBg56M2sllKgvAQp2BkO5MaO2Ww6ebs913MOkD+chJcYTVwBYy8JVCINT3rxZzl922WVlUzBC2AQQl730Ut+9lIUUVv1erwfczAcg2fv16+c9X89T9vnnn1+gLHz6vAfe85gkY0luaBRoHdV
ubFQLJp9wOzIIBfyw6OW2IS68Pnnn4tbEnWBa9966y37EvkME6DPV35w0eO6gNRBm5l1BFUlnDiXkYYmdijNHI22+VDb9Mytm4caO4UJlGlsrCBapIPfkk8J8ei3vVNFHH4kap78F5eiAH5SeduVpawEHF9zuC9BaQEHV
KD0tHtPXws44Puaw30JSgs44Aelp917+lrAAdVXHO5LUFrAAT8oPe3e09cCDvi+5nBfgtICDvhB6Wn3nr4WcMD3NUdmf2FWmD9HRbeAA37RbZQRVx7kx+K8GQVgJ8dFd4CDviFt0VanyUeh0jNq666SqI0CXdmV
S2Bd46it4ADfvS2SfszRIKyEIUuJmXKIPH+iAolgA6mcBS5BRzwI7dL2vKlnSyNCDhbdDrZxa9E+XpKHILOOBHbpe0V
ZVQZsKIWeiiNcjK7LkI0rv5cvX94cddRRstlqbV8mhRVzwE9h48f7aNKZkGqETAKkdo4slq9Tp4655JLlTN26db1zMAKLXlFVcf77Gy73gEVA3qUBSPfV+9rPV
dZZddPGADelaK9e7dW5YqYtCSBhHAK2OQSe2ZZ55xbs6wfnbAD2uQdPuKafvqq6+K2qJg5ohKw7LHkSNHGvR9JvYZZGxTYIc0l6QpJMuDoV+3gAP+V9si7T6RBYE1v6QxtEGPvk5GtyVLhSoMxNYJK0l5YjeQ77PM8880V
n3rdZywLcal10+4oYkOxoSxcGrR7w4ZEwojAAVaU9lOq3AX5y8biZ3Z0t54BfGIJSdl7cnaQJL+mApcjRixZhsjVWRip1GVysKF3P8DHE+Qmtna2nAN
```

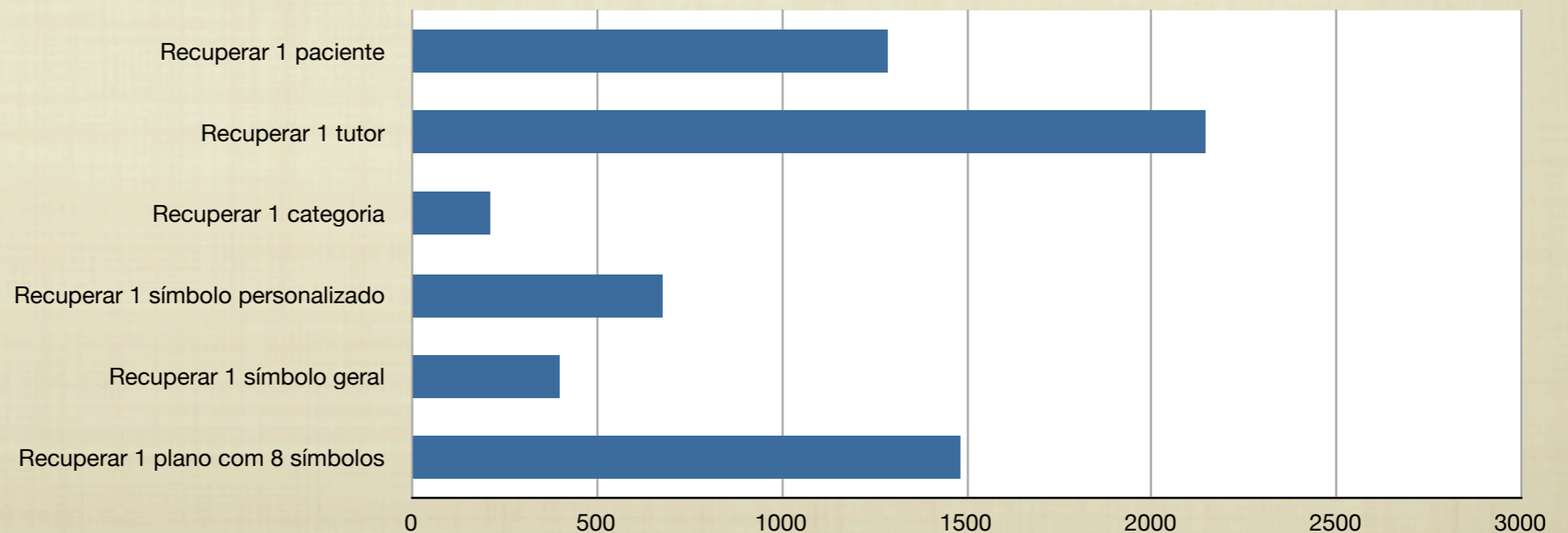
IMPLEMENTAÇÃO

RESULTADOS E DISCUSSÃO - DESEMPENHO

Operação Save



Operação Fetch



em microsegundos

IMPLEMENTAÇÃO

RESULTADOS E DISCUSSÃO - COMPARAÇÃO COM OS TRABALHOS CORRELATOS

Características / Trabalhos Correlatos	Tagarela	Projeto AMPLISOFT	Alexicom AAC
criação de símbolos personalizados	X	X	X
associação de áudio aos símbolos	X		X
troca de mensagens entre as pessoas envolvidas	X		
criação de plano de atividades	X		
possibilidade de impressão das pranchas	X	X	
histórico de observações do paciente	X		
histórico de uso dos símbolos	X		
criação de símbolos via plataforma web			X

IMPLEMENTAÇÃO

RESULTADOS E DISCUSSÃO - ELABORAÇÃO DOS REQUISITOS

- Entrevistas com o fonoaudiólogo Rodrigo França e com o tutor Dalton Solano dos Reis para:
 1. Identificar as necessidades do futuro usuário
 2. Analisar os trabalhos correlatos
 3. Analisar as ferramentas de comunicação alternativa manuais
 4. Definir a fase inicial de testes do aplicativo
- Resultado final: <http://www.youtube.com/watch?v=a0Yalc6TXSo>

CONCLUSÃO

- Objetivos alcançados com sucesso
- Tendência de melhora no processo de comunicação do paciente através do uso do aplicativo ao decorrer do tempo
- Limitações do aplicativo serão contornadas no futuro, com orientação dos usuários (tutores) e do fonoaudiólogo Rodrigo França
- Ferramentas da plataforma iOS (iOS Simulator e Instruments) se mostraram importantes no decorrer do desenvolvimento do trabalho
- Documentação: <https://www.box.com/s/1anvyhv93bjhaehshuxh/>

CONCLUSÃO

EXTENSÕES

- Implementar um servidor para realizar a troca de mensagens entre as pessoas envolvidas
- Permitir que o paciente faça a movimentação dos símbolos nas pranchas
- Permitir que o fonoaudiólogo crie pranchas dinâmicas, com número variável de símbolos
- Permitir que o fonoaudiólogo crie um plano via plataforma web
- Analisar a possibilidade de implementar outras formas de interação com os símbolos

DEMONSTRAÇÃO

Símbolos (punto)

- ZUMBA
- Audio
- NOME
- Significado
- ESCALA

SYMBOL MANAGER

- Categorías
- NOME
- COL: PALETTE
- Lista símbolos
- Personalizado

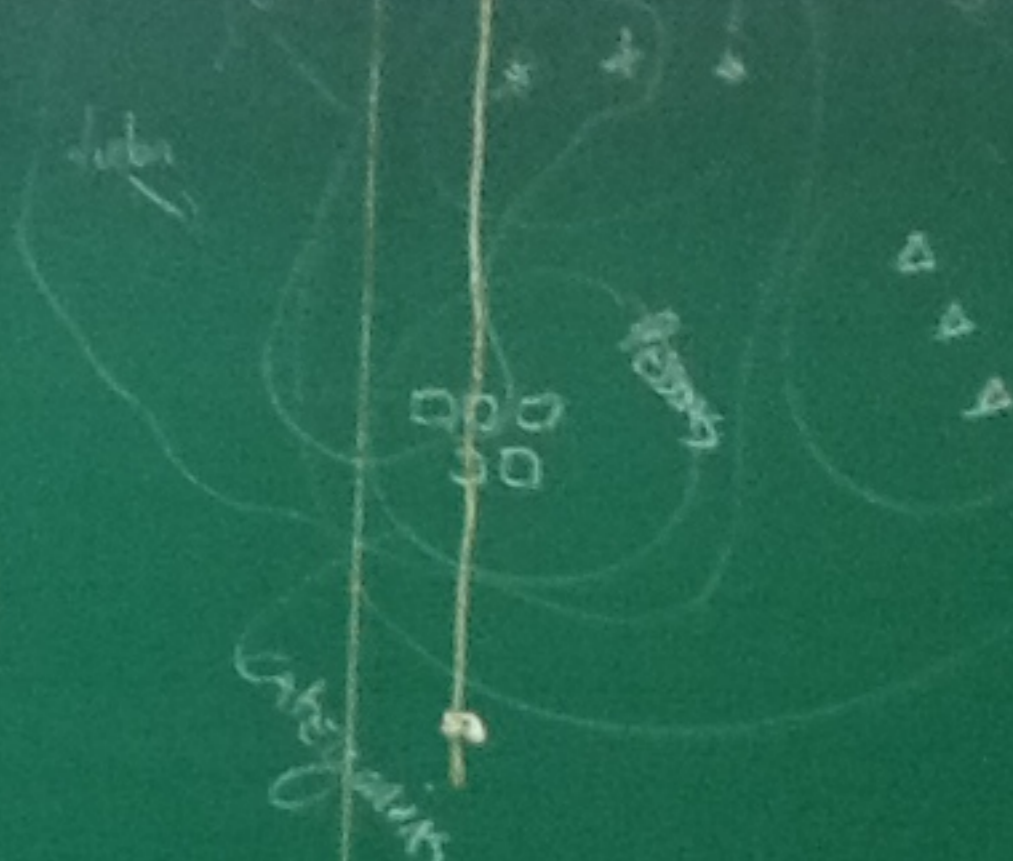
Reservas

- BEER
- PIZZA
- DANISH
- TOMATO SAUCE


Artista

Walter (foto)
ANA (AVA)




MANEJO DE CATEGORIAS




Saídas



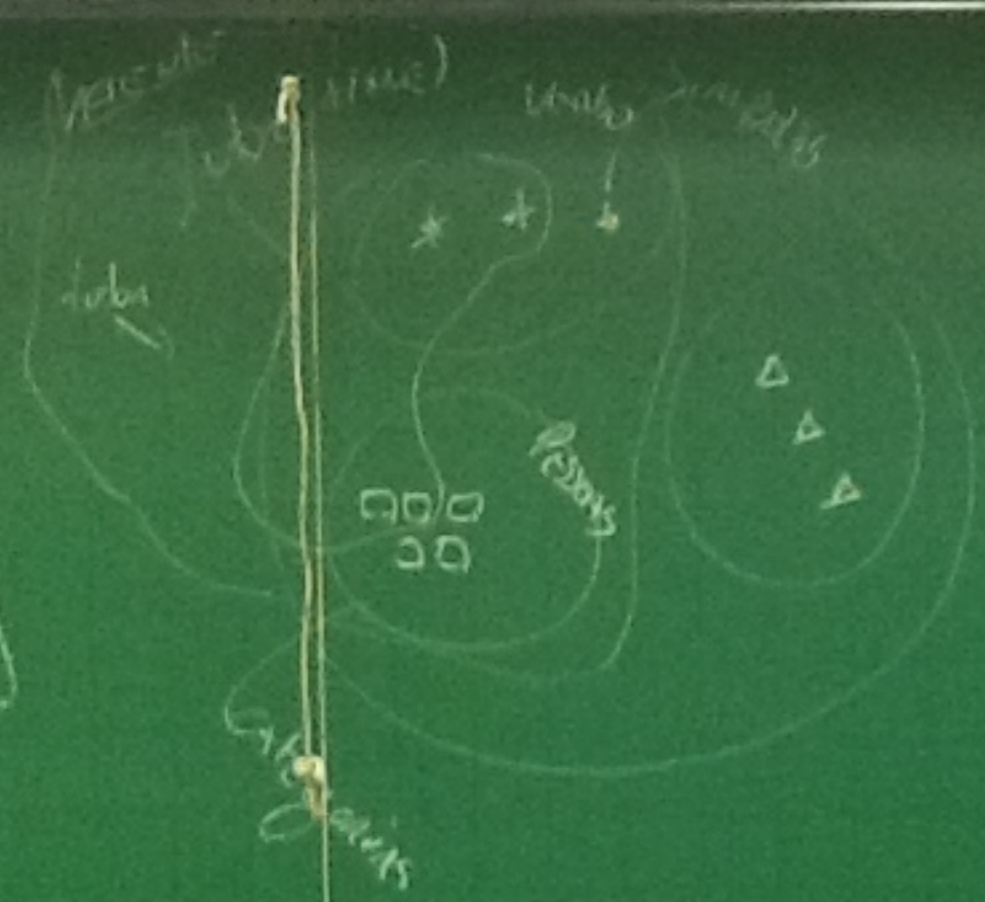
 Pano/PAC → PCA GRANDE: Impresso, Escultido, Tutora
 (Lubrificação)

Faml/PAC → PANCHAS (ideação sigab): Imp, Escultidas
 LAY-out -   

PANO  → História (Pano) + cada 5000 (organizado por PATA)

PANO → Tutora (Manual) cada ← PANO GRANDE PANCHAS

lista
 -
 -
 -
 -



Atores

① FUNDO (ADM)

② Cadastram Paciente

- CRIA OS SÍM/CATEGORIAS

{ Geral (FOM)
& Personalizado (Paciente)

③ Cadastram tutores

④ definir "Planos" tutores

{ DA GEMME
SIMILARES

▷ Tutores (PAP, EDD, PAI, ...
(AMBIENTE: LOCAL)

ALME
JUNIFER
etc

▷ PACIENTE

— TUTOR

— PLANO

Simb Simb (COR)

— MANCHA → "simon"

DATA

▷ CADASTRO FUNDOS
SINHA

Pessoa

NOME
DATA NASC.
E-MAIL

FUNÇÃO

PACIENTE

Patologia (CROSS)

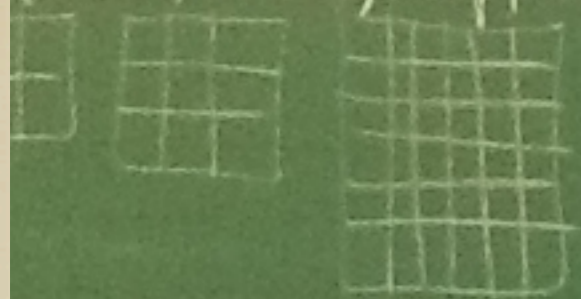
Historico (DATA)

INICIO USO

FIM USO

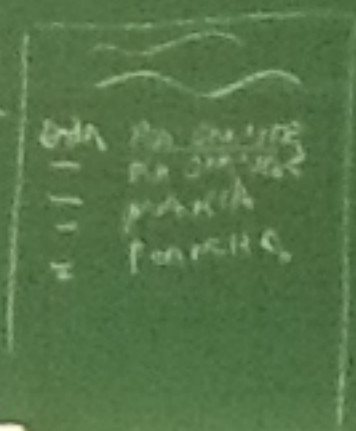
de: Impresso, Escrito, Tutor

(CATEGORIA SÍMBOLO): Imp, Escritas



A cada sessão (dependendo da data)

DATA ← PENSamentos
Pensamentos



1) SediMENTAR Símbolo (RA). Espaço Alguas
RA's em locais

2) PANCHAS

3) CRIA DINAMICAMENTE PANCHAS

Resumidamente
? (DATA (SÍMBOLO): PENSAMENTOS)
? → CATEGORIAS FACILITADAS
? → ANCIEN

```
- (void)startRecordingWithFilePath:(NSString*)recorderFilePath
{
    AVAudioSession *audioSession = [AVAudioSession sharedInstance];
    NSError *error;

    [audioSession setCategory:AVAudioSessionCategoryPlayAndRecord error:&error];
    [audioSession setActive:YES error:&error];

    NSMutableDictionary *recordSettings = [[NSMutableDictionary alloc] init];
    [recordSettings setValue:[NSNumber numberWithInt:kAudioFormatAppleIMA4] forKey:AVFormatIDKey];
    [recordSettings setValue:[NSNumber numberWithFloat:12000.0] forKey:AVSampleRateKey];
    [recordSettings setValue:[NSNumber numberWithInt:1] forKey:AVNumberOfChannelsKey];

    NSURL *url = [NSURL fileURLWithPath:recorderFilePath];
    NSData *audioData = [NSData dataWithContentsOfFile:[url path] options:0 error:&error];

    if (audioData) {
        NSFileManager *fm = [NSFileManager defaultManager];
        [fm removeItemAtPath:[url path] error:&error];
    }

    recorder = [[AVAudioRecorder alloc] initWithURL:url settings:recordSettings error:&error];
    [recorder setDelegate:self];
    [recorder prepareToRecord];
    [recorder setMeteringEnabled:YES];
    [recorder recordForDuration:(NSTimeInterval)5];
}
```

```

dispatch_async(dispatch_get_main_queue(), ^{
    NSString *iCloudEnabledAppID = @"49WEM39FE7.com.fabeni.pca";
    NSString *dataFileName = @"MyModel.sqlite";
    NSString *iCloudDataDirectoryName = @"Data.nosync";
    NSString *iCloudLogsDirectoryName = @"Logs";
    NSFileManager *fileManager = [NSFileManager defaultManager];
    NSURL *localStorage = [[self applicationDocumentsDirectory]URLByAppendingPathComponent:dataFileName];
    NSURL *iCloud = [fileManager URLForUbiquityContainerIdentifier:nil];

    if (iCloud) {
        NSURL *iCloudLogsPath = [NSURL fileURLWithPath:[iCloud path]stringByAppendingPathComponent:iCloudLogsDirectoryName]];

        if ([fileManager fileExistsAtPath:[iCloud path]stringByAppendingPathComponent:iCloudDataDirectoryName] == NO) {
            NSError *fileSystemError;
            [fileManager createDirectoryAtPath:[iCloud path] stringByAppendingPathComponent:iCloudDataDirectoryName]
                withIntermediateDirectories:YES
                attributes:nil
                error:&fileSystemError];
        }

        NSString *iCloudData = [[[iCloud
            path]stringByAppendingPathComponent:iCloudDataDirectoryName]stringByAppendingPathComponent:dataFileName];

        NSMutableDictionary *options = [NSMutableDictionary dictionary];
        [options setObject:[NSNumber numberWithInt:YES] forKey:NSMigratePersistentStoresAutomaticallyOption];
        [options setObject:[NSNumber numberWithInt:YES] forKey:NSInferMappingModelAutomaticallyOption];
        [options setObject:iCloudEnabledAppID forKey:NSPersistentStoreUbiquitousContentNameKey];
        [options setObject:iCloudLogsPath forKey:NSPersistentStoreUbiquitousContentURLKey];

        [psc lock];

        [psc addPersistentStoreWithType:NSSQLiteStoreType
            configuration:nil
            URL:[NSURL fileURLWithPath:iCloudData]
            options:options
            error:nil];

        [psc unlock];
    }
}

```

```
- (BOOL)createSymbol:(Symbol**)symbol withName:(NSString*)name andImage:(UIImage*)image andMeaning:(NSString*)meaning andSound:
  (NSString*)sound andCategory:(Category*)category andPatient:(Patient*)patient
{
    NSError *error;
    NSData *imageData = UIImagePNGRepresentation(image);

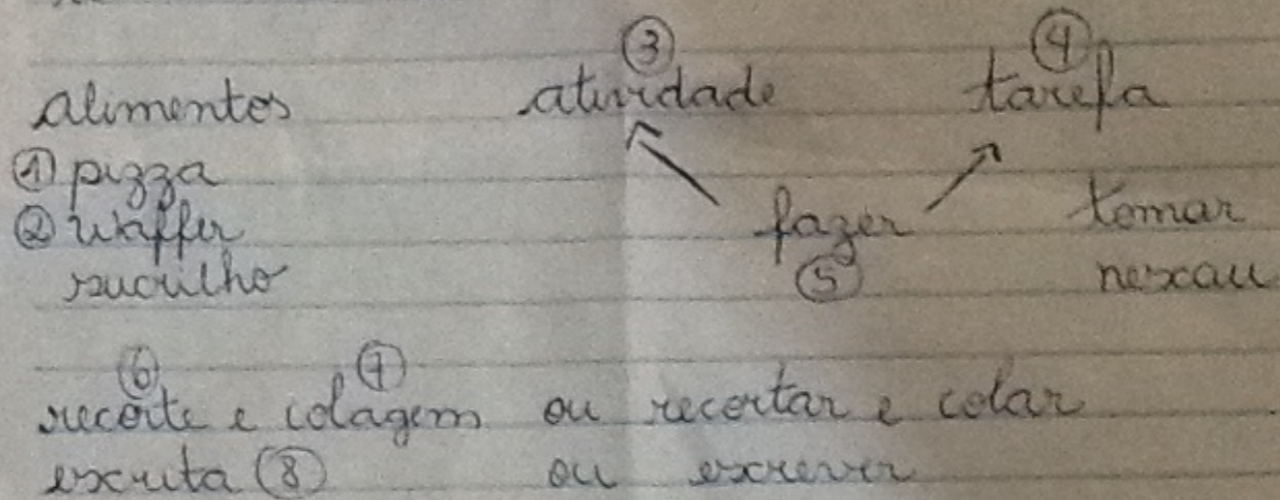
    [*symbol setSymbolName:name];
    [*symbol setSymbolImage:imageData];
    [*symbol setSymbolMeaning:meaning];
    [*symbol setSymbolSound:sound];
    [*symbol setSymbolCategory:category];

    if (patient) {
        NSMutableSet *patientsSet = [NSMutableSet setWithSet:[*symbol patients]];
        [patientsSet addObject:patient];
        [*symbol setPatients:patientsSet];
        [*symbol setSymbolIsGeneral:[NSNumber numberWithInt:NO]];
    } else {
        [*symbol setSymbolIsGeneral:[NSNumber numberWithInt:YES]];
    }

    if (![managedObjectContext save:&error]) {
        return NO;
    }
    return YES;
}
```


Dous pais!

Segue abaixo alguns itens que podemos acrescentar no CAA



Atenciosamente

Professora Aline

04/06/12

to Fernanda Xavier

Ola Aline e Fernanda!

Para o item "tomar", usamos "beber"
e daí usaremos outra figura para "rescau".

Att, Rodrigo 05/06/2012

Disciplinas

Ensino Religioso / Biblioteca

Antes

Informática

Educação Física

OK

As professora

05/06

- Core Audio Format (.caf)
- Desenvolvido pela Apple
- Projetado para superar as limitações dos antigos formatos de áudio digital, incluindo AIFF e WAV
- Não se limita a um tamanho de arquivo de 4 GB
- Teoricamente pode salvar centenas de anos de áudio gravado devido ao seu offset de 64 bits
- IMA4 é um formato de compressão, utilizado para comprimir os arquivos .caf
- Compressão de 4:1