

# UNASP

**CENTRO UNIVERSITÁRIO ADVENTISTA DE SÃO PAULO**  
**CAMPUS SÃO PAULO**  
**CURSO DE CIÊNCIA DA COMPUTAÇÃO**



**PROJETO DE TRABALHO DE CONCLUSÃO DE CURSO**

## **Ambiente baseado em comando de voz destinado a pessoas com limitação motora**

Aluno

Max Felipe Machado Bezerra

Orientador

Cristiano Felício

São Paulo, SP, Novembro de 2015

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>2</b>	<b>OBJETIVOS</b>	<b>2</b>
2.1	Objetivo Geral	2
2.2	Objetivos Específicos	2
<b>3</b>	<b>JUSTIFICATIVA</b>	<b>3</b>
<b>4</b>	<b>REFERENCIAL TEÓRICO</b>	<b>4</b>
4.1	IDE	4
4.1.1	Visual Studio	4
4.1.2	Eclipse	5
4.1.3	NetBeans	6
4.1.4	XCode	7
4.1.5	InGriDE	8
4.1.6	IDEs adaptadas para deficientes motores	9
4.2	Reconhecimento da Fala	10
4.2.1	Microsoft Speech	11
4.2.2	Java Speech API	11
4.2.3	Web Speech API	12
4.2.4	VoiceXML	12
4.2.5	Siri	13
4.2.6	Voice Note II	13
<b>5</b>	<b>METODOLOGIA</b>	<b>14</b>
<b>6</b>	<b>VISÃO GERAL DO PROJETO</b>	<b>15</b>
<b>7</b>	<b>RECURSOS E VIABILIDADE</b>	<b>16</b>
<b>8</b>	<b>CRONOGRAMA</b>	<b>16</b>
<b>9</b>	<b>REFERÊNCIAS</b>	<b>17</b>

## ÍNDICE DE FIGURAS

<b>Figura 1 – Microsoft Visual Studio 2015</b> .....	<b>05</b>
<b>Figura 2 – Eclipse IDE</b> .....	<b>06</b>
<b>Figura 3 – NetBeans IDE 6.8</b> .....	<b>07</b>
<b>Figura 4 – XCode IDE</b> .....	<b>08</b>
<b>Figura 5 – InGriDE IDE</b> .....	<b>09</b>
<b>Figura 6 – Assistente Pessoal Siri</b> .....	<b>13</b>
<b>Figura 7 – Visão Geral do Projeto</b> .....	<b>15</b>

## 1 Introdução

Segundo estimativas da Organização Mundial da Saúde (OMS), em países desenvolvidos, cerca de 10% da população possui algum tipo de deficiência. Já em países em desenvolvimento, é provável que 12 a 15% das pessoas tenha alguma deficiência. Entre esses, é provável cerca de 20% possui algum tipo de deficiência física. Levando em consideração o total dos portadores de qualquer deficiência, somente 2% deles tem atendimento especializado, público ou privado. (SEESP/MEC, 2006).

Levando em consideração que atualmente o mundo é coordenado através da Tecnologia da Informação, torna-se necessária a inclusão abrangente a todos. Porém, apesar de todos os recursos e da popularização da inclusão digital, ainda existem muitas restrições referentes a determinados tipos de necessidades especiais. A partir disso, pode-se dizer de que não adianta em nada a infinidade de recursos, se não houver dispositivos e aplicações específicos que supram essas necessidades.

Segundo LEMOS (2003, p13), vivemos o tempo da *cybercultura*, não como futuro e sim como presente, e a mesma representa a cultura contemporânea, que se torna consequência direta da evolução da cultura técnica moderna. Partindo desse pressuposto, não saber ou não poder manipular os recursos tecnológicos é coeficiente direto da exclusão da cultura atual.

LEMOS (2003, p15) também cita que a *cybercultura* também é responsável, através de mídias sociais, por de alguma forma buscar o sentimento de religação com as outras pessoas conectadas. Esse sentimento de religação é notado através do uso da Tecnologia Assistiva por pessoas com deficiência. Através do uso de aplicativos e ferramentas tecnológicas, pode-se fazer com que o deficiente se sinta ainda mais atuante e capacitado para interação com o mundo.

A partir dos fatos supra mencionados, este trabalho propõe o desenvolvimento de Ambiente baseado em comando de voz destinado a pessoas com limitação motora, que possibilitará a introdução didática ao desenvolvimento de *software* das mesmas.

## 2 Objetivos

### 2.1 Objetivo Geral

O objetivo desse trabalho é implementar um ambiente baseado em comando de voz para pessoas com limitação motora.

### 2.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- Realizar o levantamento bibliográfico e estudo de trabalhos correlatos;
- Criar uma interface com comando de voz;
- Estimular a inclusão digital de pessoas com dificuldade de acesso pelos meios tradicionais desenvolvimento de *software*;

### 3 Justificativa

No cenário atual de desenvolvimento de *software*, vê-se um grande aumento de desenvolvedores “informais” entre os meios de jogos e aplicativos *mobile*. É muito comum encontrar cada vez mais aventureiros que fazem grande sucesso na área de programação. Pode-se dizer que um dos fatores desse grande aumento de programadores no mercado, se deve a acessibilidade a tutoriais, e interfaces de desenvolvimento mais intuitivas, bem como a maior divulgação e incentivo da área por grandes empresas e meios de comunicação.

Hoje, o desenvolvimento de *software* é parte essencial em todas as empresas de tecnologia, fazendo assim com que o mercado de trabalho esteja sempre em expansão. Mesclando essas grandes mudanças com a integração de deficientes no mercado, vê-se a grande necessidade de uma plataforma de desenvolvimento não só intuitiva, como também adaptada para pessoas com necessidades especiais.

Este projeto tem o objetivo de criar um ambiente baseado em comando de voz voltado para pessoas como limitação motora, tendo em vista a iniciação e introdução didática dessas pessoas no mundo desenvolvimento de *software* e inclusão equivalente a outras pessoas na área da tecnologia e programação.

## 4 Referencial Teórico

### 4.1 IDE

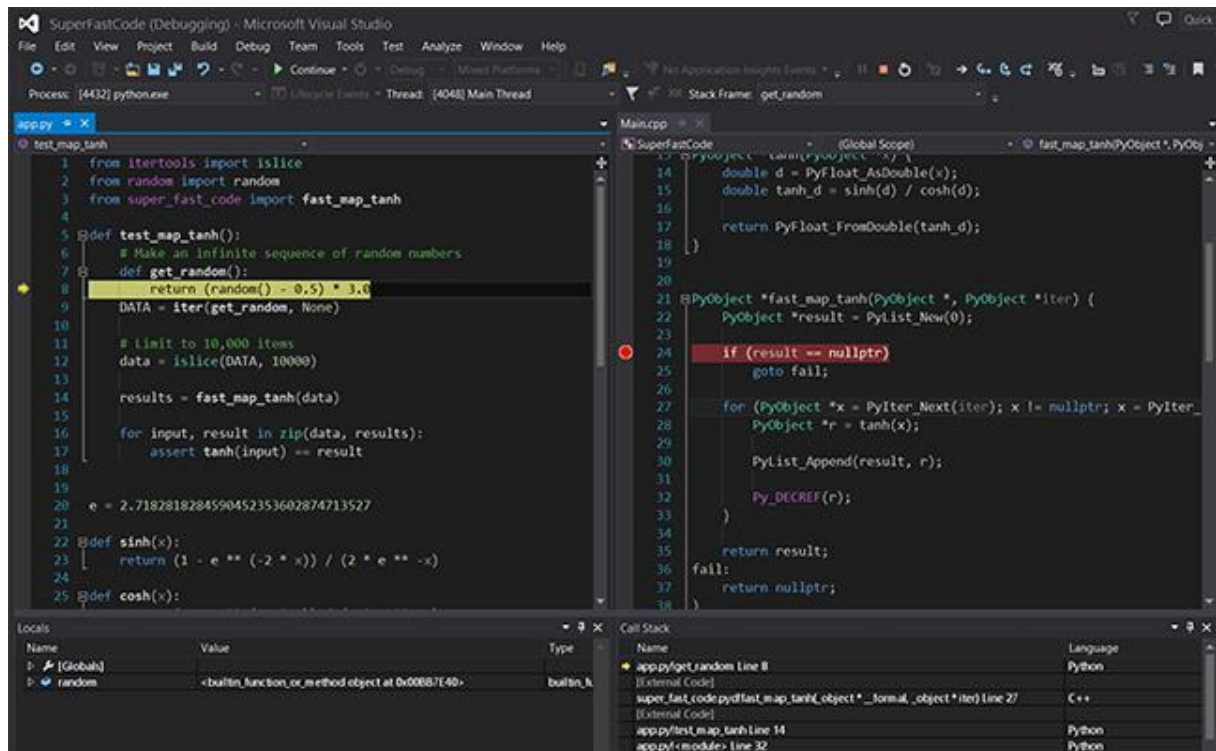
Segundo SEBESTA (2011, p. 52), um ambiente integrado de desenvolvimento (conhecido como IDE, de "*integrated development environment*") é um conjunto de ferramentas que auxiliam no desenvolvimento de *software*. O ide disponibiliza como ferramentas um compilador, um editor de código, ferramentas que auxiliam no desenvolvimento de sistemas e um depurador. Geralmente cada ferramenta pode ser acessada por uma interface de usuário, trazendo assim uma melhora no desenvolvimento e manutenção do *software*.

#### 4.1.1 Visual Studio

Um dos principais IDE do mercado, o *Visual Studio* (figura 1) é composto por um conjunto de ferramentas para desenvolvimento de aplicativos *Desktop* e *Web*. É possível criar aplicativos individuais e desenvolver e implantar soluções empresariais com suas ferramentas de desenvolvimento baseadas em componentes próprios e outras tecnologias (MICROSOFT, 2015).

De acordo com SEBESTA (2011, p.52) o *Visual Studio* é relativamente recente na evolução dos IDE's, com sua interface baseada em janelas.

Figura 1 – Microsoft Visual Studio 2015



Fonte: VISUALSTUDIO (2015)

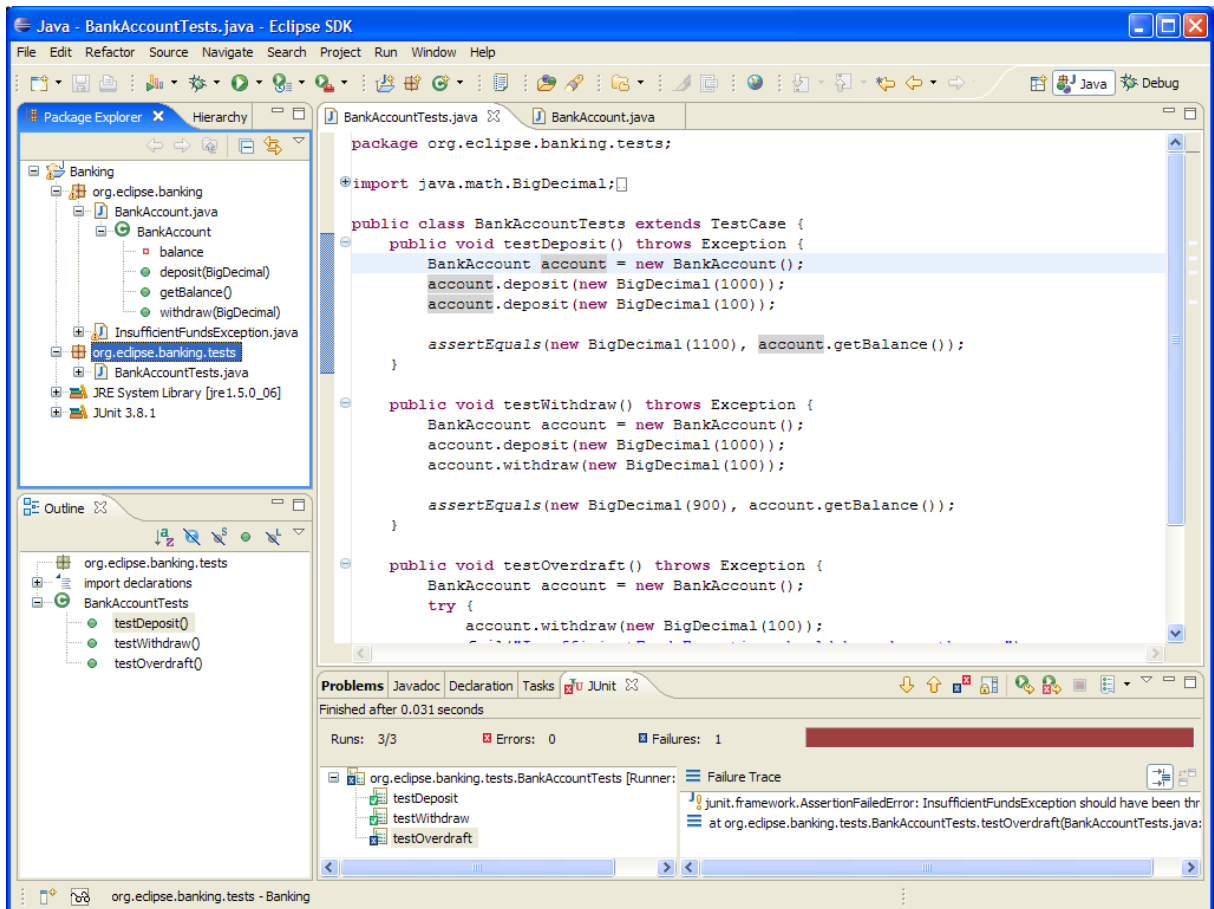
#### 4.1.2 Eclipse

O Eclipse foi criado pela IBM em 2001, sendo uma plataforma de desenvolvimento de *software* livre e extensível, trazendo conjuntos de *plug-ins* e ferramentas para desenvolvimento. Em 2004 foi criada a Eclipse *Foundation* para se manter como organizadora da comunidade Eclipse, sendo a mesma sem fins lucrativos e independente (ANISZCZYK e GALLARDO, 2012).

O Eclipse IDE (figura 2), hoje o IDE Java mais utilizado no mundo, compreende vários tipos de linguagem e *plug-ins* para emulação do desenvolvimento da plataforma, trazendo com uma das principais vantagens faz uso do *SWT* e com seu desenvolvimento baseando-se em *plug-ins* que auxiliam em suas necessidades, trazendo um grande ganho de suporte para o desenvolvedor (PALMEIRA, 2012).



Figura 2 – Eclipse IDE



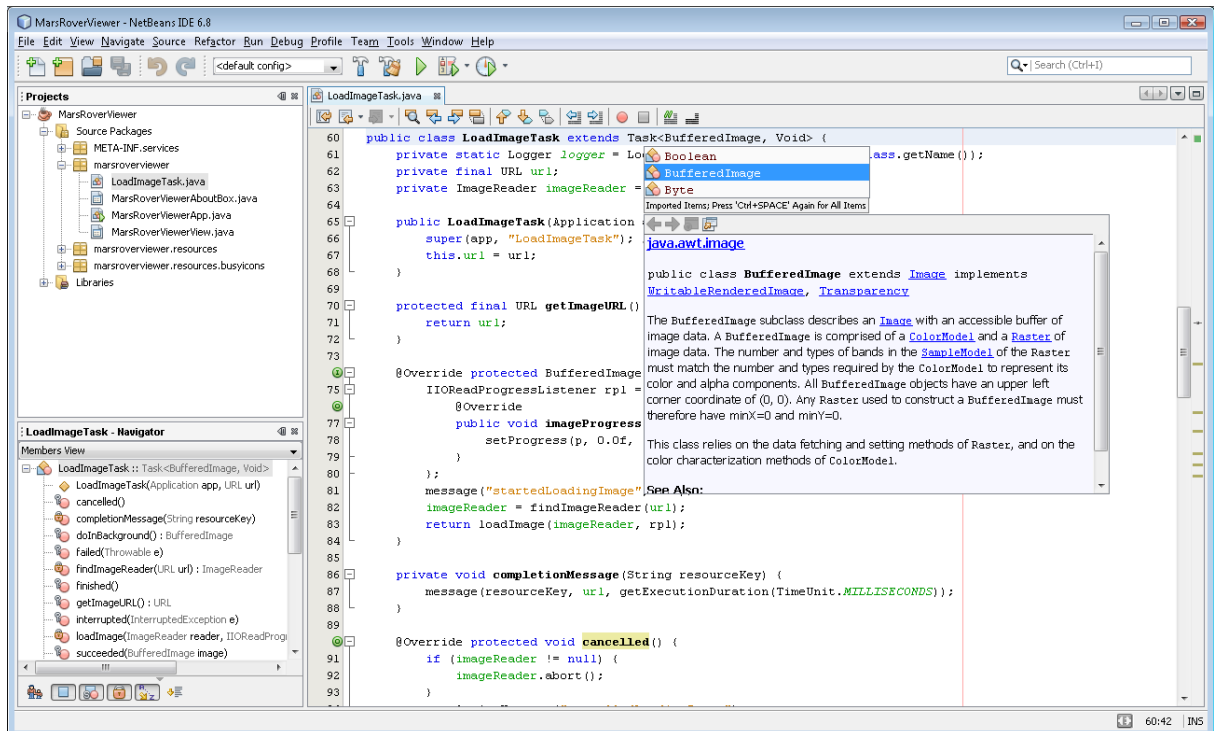
Fonte: Eclipse (2015)

#### 4.1.3 NetBeans

O *NetBeans* (figura 3) é um IDE livre e sem restrições em sua utilização, que foi escrito em Java mas pode suporta todas as linguagens de programação. Além disso, existem módulos no *NetBeans* que estendem suas funcionalidades (NETBEANS, 2015).

SEBESTA (2011, p.52) diz que o *NetBeans* tem como prioridade o desenvolvimento de *softwares web* utilizando a linguagem Java, porém oferece suporte a linguagens como Ruby, PHP e *Javascript*. O *NetBeans*, bem como outros ambientes integrados de desenvolvimento também é um *framework*.

Figura 3 – NetBeans IDE 6.8



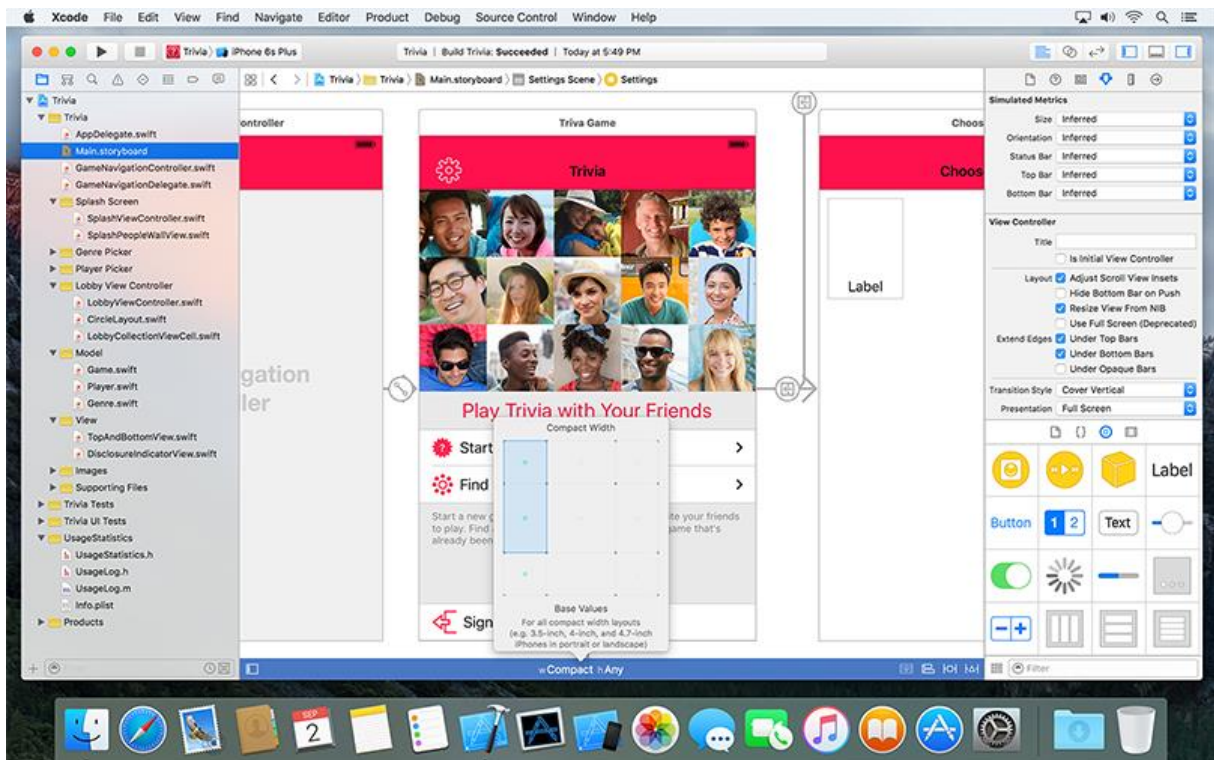
Fonte: NETBEANS (2015)

#### 4.1.4 XCode

O ambiente integrado de desenvolvimento *XCode* (figura 4) é utilizado para o desenvolvimento de aplicações para o sistema operacional iOS (iPad/iPod/iPhone) e Mac OS X. O IDE é mantido pela Apple e disponibilizado no pacote de ferramentas de desenvolvimento, e anteriormente era conhecido como *Project Builder*. Em diferentes sistemas operacionais o *XCode* pode ser utilizado através de máquina virtual (NETO et al., 2014).

Com o *XCode*, além de desenvolver aplicações iOS e analisar as mesmas, também executam-se testes usando simuladores dos dispositivos e também se analisa o desempenho de algumas das funcionalidades presentes (SILVA e SILVEIRA, 2011).

Figura 4 – XCode IDE



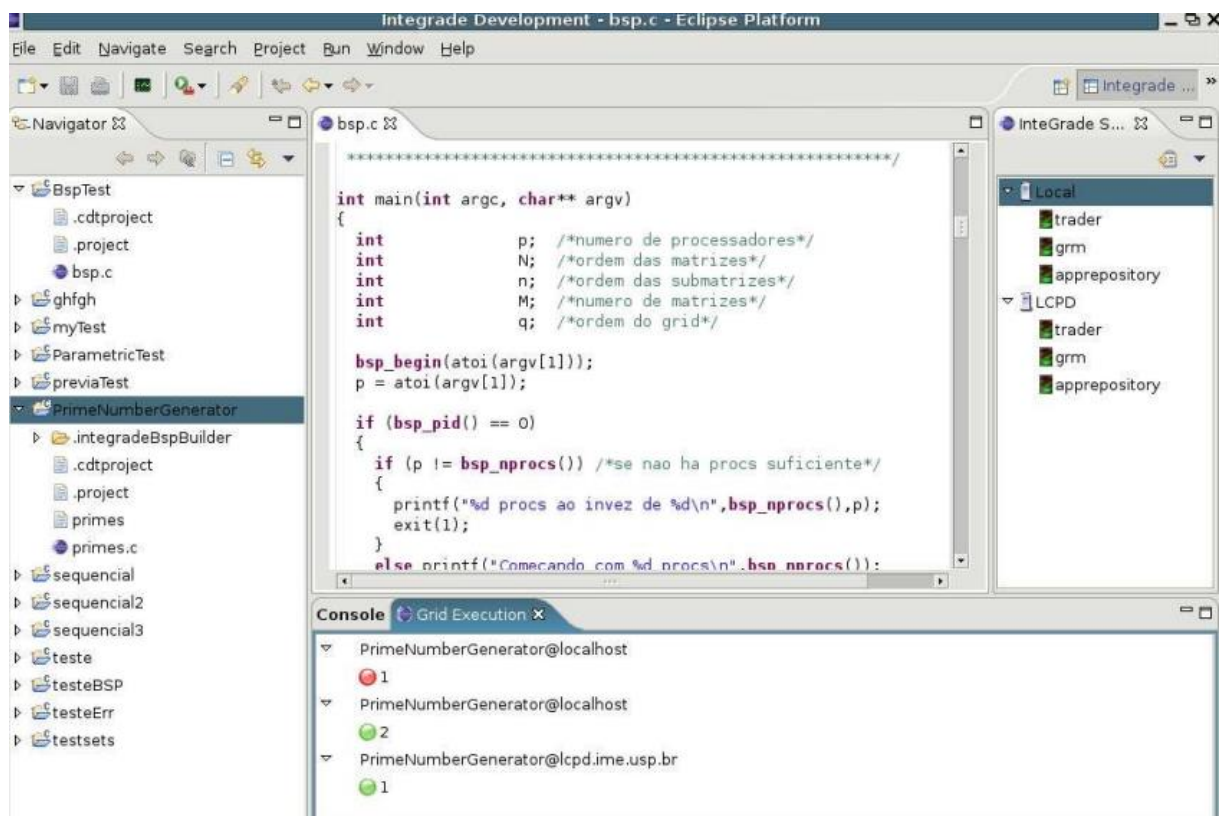
Fonte: APPLE (2015)

#### 4.1.5 InGriDE

Segundo GUERRA (2007) InGriDE é um ambiente integrado e extensível de desenvolvimento, desenvolvido com base na plataforma Eclipse e criado para diminuir as dificuldades de desenvolvimento de aplicações em Grade, provendo ferramentas únicas compatíveis com diferentes sistemas de *middleware*, sendo criadas baseando-se na interface de programação GAT (*Grid Application Toolkit*).

Uma grade computacional é uma infra-estrutura de *software* capaz de interligar e gerenciar diversos recursos distribuídos geograficamente, com o objetivo de oferecer ao usuário da grade acesso transparente a estes recursos, independente de sua localização. Os avanços tanto na implementação quanto na padronização das grades computacionais vêm alcançando um nível de robustez que possibilita a implantação de ambientes de produção de qualidade na comunidade de pesquisa acadêmica e desperta um grande interesse da indústria.

Figura 5 – InGriDE IDE



#### 4.1.6 IDEs adaptadas para deficientes motores

Atualmente, pessoas com algum tipo de deficiência utilizam recursos disponibilizados para sistemas operacionais, como por exemplo, aplicativos (nativos ou não) de leitura de texto ou de reconhecimento de comandos por voz para realizar tarefas básicas, e também para auxiliar no desenvolvimento de *software*. Porém, de acordo com pesquisas recentes, ainda não há um IDE que disponibilize recursos para desenvolvimento de sistemas apenas por comando de voz, o que dificulta a iniciação de pessoas com deficiência motora no mundo da programação.

## 4.2 Reconhecimento da Fala

Geralmente a expressão “reconhecimento de voz” é usada com vários sentidos, sendo que na verdade se referem a diferentes tecnologias. As principais áreas que o processamento de voz pode ser categorizado são: síntese de voz, comando de voz, fala natural e autenticação de voz. O comando de voz é responsável por processar um trecho da fala, geralmente isso indica um tipo de ação que o sistema deve obedecer. Como o sistema já tem os comandos disponíveis, o processamento se torna mais simples. O reconhecimento da fala contínua se dá por várias palavras que tenham algum sentido semântico e assim é convertido para texto. Já a síntese de voz faz o oposto do reconhecimento de fala, pois obtém o texto em formato digital e fazendo assim a leitura em voz alta. No caso da autenticação por voz o sistema reconhece a voz de uma pessoa, tendo em vista que cada voz é diferente proporcionando assim fazer autenticações pessoais por meio da voz (GUILHOTO e ROSA, 2001).

O reconhecimento de voz evoluiu bastante ao longo dos últimos anos. Inicialmente, o reconhecimento processava-se apenas em modo discreto, isto é, o utilizador tinha que fazer uma pausa entre cada palavra ditada. Actualmente, o utilizador já tem a possibilidade de efectuar ditados contínuos ao computador. Para além disso, o reconhecimento também se tornou mais inteligente, uma vez que possui um conjunto de regras gramaticais incorporadas, permitindo assim perceber melhor o que está a ser ditado. O reconhecimento de voz usa diferentes técnicas para reconhecer a voz humana. Funciona assim como uma “*pipeline*” que transforma os sinais de áudio digitais provenientes da placa de som em fala reconhecida. Estes sinais passam por diversas etapas, ao longo das quais são aplicados métodos matemáticos e estatísticos de forma a tentar compreender o que está a ser ditado.

Os sistemas de reconhecimento de fala contínua se baseiam em principalmente no reconhecimento estatístico de padrões. Estes sistemas têm como objetivo criar uma função de probabilidade que seja capaz de mapear amostras de sinal de fala, que são representados por sequência de vetores acústicos (MORAIS, 1997).

#### 4.2.1 Microsoft Speech

A Microsoft *Speech* é uma plataforma que consiste em um kit de desenvolvimento (SDK), um *runtime* e pacotes de idiomas que permitem que seja feito o reconhecimento de voz ou de texto de um idioma específico onde é possível ser redistribuído por meio de aplicativos próprios. A plataforma Microsoft *Speech* SDK fornece um conjunto de ferramentas de desenvolvimento para a gestão da plataforma *Speech* em aplicativos habilitados para comando de voz. É possível reconhecer palavras e gerar voz sintetizada (*Text-To-Speech*, ou *TTS*), fazendo com possa ter uma maior interação dos seus usuários com seus aplicativos. A plataforma *Speech* SDK também inclui o *namespace* “Microsoft.Speech.VoiceXml” que dá suporte para aplicações de voz autorais que utilizam a linguagem de marcação *VoiceXML*, além de Ferramentas de Desenvolvimento Microsoft *Grammarfornecer*, que consiste em um conjunto abrangente de aplicativos de linha de comando com o qual você pode validar, analisar e ajustar suas gramáticas de reconhecimento de fala (MICROSOFT, 2015).

#### 4.2.2 Java Speech API

A API Java *Speech* foi lançada em 1998, desenvolvida pela *Sun Microsystems, Inc.*, em colaboração com as principais empresas de tecnologia de fala: *Apple Computer, Inc.*, *AT & T*, *Dragon Systems, Inc.*, *IBM Corporation*, *Novell, Inc.*, *Philips* e *Texas Instruments Incorporated*. Esta API faz parte de uma família de APIs que trabalham junto a uma suíte de ferramentas para oferecer maior capacidade de comunicação e gráficos melhorados. A API Java *Speech* permite incorporar tecnologia de fala em interfaces de usuário para seus *applets* e aplicativos baseados na tecnologia Java. Esta API especifica uma interface multi-plataforma para suportar comando e controle, sistemas de reconhecedores de ditado e sintetizadores de voz (ORACLE, 2015).

Segundo ENDEN (2001) a API Java *Speech* foi projetada para ser a camada entre o mecanismo de fala e a camada de programação de aplicativos, sendo parte das APIs Java que foram projetadas para trabalhar em conjunto com os desenvolvedores para produzir aplicações pessoais e *web* com conteúdo multimídia. Ela define um grupo de interfaces que será implementado o mecanismo de fala. A coleção de APIs no qual a Java *Speech* faz parte consiste no Java *Media Framework*, Java *Sound*, Java *Speech*, Java *Telephony*, Java 2D, 3D e Java *Advanced Imaging*.

### 4.2.3 Web Speech API

De acordo com ADORF (2013) a *Web Speech API* é uma *API JavaScript* experimental, trazendo como característica fundamental o reconhecimento de voz em si muito transparente para o desenvolvedor *web*. Esta API foi desenvolvida pela W3C, sendo aberta a todos e aparentemente sendo em grande parte conduzida por duas empresas: Google e *OpenStream*. A utilidade da *Web Speech API* está limitada pela performance do serviço *web* que executa o reconhecimento de voz.

Segundo SHIRES (2015) a nova *API JavaScript*, *Web Speech API* torna mais fácil adicionar o reconhecimento de voz em páginas *web*. Essa API disponibiliza um bom controle e flexibilidade sobre os recursos de reconhecimento de fala no navegador Google Chrome versão 25 ou posterior.

### 4.2.4 VoiceXML

De acordo com REHOR (2001), *VoiceXML* é uma linguagem para a criação de interfaces de usuário de voz, principalmente para telefones. Ele usa o reconhecimento de voz e tons DTMF e para entrada, e de áudios pré-gravados e síntese de texto-para-voz (*TTS*) para a saída. Ele é baseado em *Worldwide Web Consortium (W3C) Extensible Markup Language (XML)*, e usa do paradigma *web* para o desenvolvimento e implantação de aplicativos. Por ter uma linguagem comum, desenvolvedores de aplicativos, fornecedores de plataformas, fornecedores de ferramentas e todos podem se beneficiar da portabilidade de código e reutilização.

Segundo REHOR (2001), com *VoiceXML*, o desenvolvimento de aplicações de reconhecimento de voz é bastante simplificado, utilizando uma infra-estrutura *web* familiarizada, incluindo as ferramentas e servidores da *Web*. Ao invés de usar um *PC* com um navegador *Web*, qualquer telefone pode acessar aplicações *VoiceXML* através de um "intérprete" *VoiceXML* (também conhecido como um "navegador") rodando em um servidor de telefonia. Considerando que o HTML é comumente usado para a criação de aplicações *Web* gráficas, *VoiceXML* pode ser usado para aplicativos *Web* ativados por voz.

#### 4.2.5 Siri

O Siri é um aplicativo para iOS do tipo assistente pessoal, que usa processamento de linguagem natural para executar ações. Com ele é possível enviar mensagens, responder perguntas ou abrir aplicativos, dentre outras opções. Ele traz consigo um *HomeKit* que torna possível deixar sua voz ser o controle remoto para produtos conectados em sua casa. O Siri trabalha com serviços *on-line* para ampliar sua capacidade de resposta, serviços esses como: *Wikipedia*, *Yelp*, *Rotten Tomatoes*, *Shazam* e outros (APPLE, 2015).

**Figura 6 – Assistente Pessoal Siri**



**Fonte: APPLE (2015)**

#### 4.2.6 Voice Note II

*Voice Note II* é um aplicativo de reconhecimento de fala para o navegador Google Chrome. É necessário ter conexão à internet e microfone para que o aplicativo funcione no modo de reconhecimento de fala, mas sem conexão com a internet é possível usá-lo como bloco de notas e dicionário. Com o *Voice Note* é possível escrever textos rapidamente apenas com a fala. Tendo suporte para várias línguas (inclusive o português), é possível criar dicionários com palavras que poderão substituir as que foram faladas (VOICENOTE, 2015).



## 5 Metodologia

Para a realização desse projeto, será feita uma pesquisa bibliográfica a respeito de IDEs e comando de voz. Também serão realizados estudos e testes de ferramentas, bem como de bibliotecas de comando de voz para o sistema operacional Windows.

Será desenvolvido o protótipo do IDE proposto (parte visual), integrando com uma biblioteca de comando de voz. Após o desenvolvimento e integração, será criada uma interface para receber o comando de voz.

Ocorrerão testes de validação no sistema por usuários pertencentes ao grupo alvo. Após estes passos, será feita a conclusão e entrega do trabalho.

## 6 Visão Geral do Projeto

Este projeto (Figura 7) tem como objetivo criar um ambiente baseado em comando de voz onde uma pessoa com limitação motora irá emitir o comando de voz e um IDE especial reconhecerá o que está sendo falado e registrará como código em tela. A parte do projeto denominada “Narrador de Código” já foi realizada anteriormente e proporcionará uma integração. Como projetos futuros, desejamos incorporar esse IDE com um compilador, ele também ativará uma máquina virtual, fazendo assim o processamento e execução do código que foi ditado. No ato da execução, existirá um narrador de código que narrará o que estará sendo executado para o programador com limitação motora.

**Figura 7 – Visão Geral do Projeto**



## 7 Recursos e Viabilidade

Recursos necessários para a realização do projeto:

- Recursos de Hardware: Será necessário um Microcomputador PC ou compatível com sistema operacional Windows, processador Intel-i3/AMD-PhenomX4 ou superior com 2GB de RAM e conexão à internet.
- Recursos de Software: Ambientes integrados de Desenvolvimento, Bibliotecas de Reconhecimento de Voz para integração e Navegador Google Chrome atualizado.
- Recursos Humanos: Ter um grupo de pessoas portadoras de necessidades especiais para testar e avaliar a aplicação desenvolvida. Existe, nesta instituição um projeto de extensão voltado para pessoas com esse perfil. A proposta então é fazer um trabalho conjunto para alcançar os resultados desejados.

## 8 Cronograma

O cronograma mostra as atividades que devem ser realizadas para atingir o objetivo proposto. (em função de 3 semestres)

Tabela 1 - Cronograma

	2015				2016									
Atividades	Ago	Set	Out	Nov	Fev	Mar	Abr	Mai	Jun	Ago	Set	Out	Nov	
1. Tema e Orientador														
2. Pesquisa Bibliográfica e Leitura														
3. Apresentação PTCC														
4. Estudo de ferramentas de desenvolvimento														
5. Estudo de bibliotecas de comando de voz para Windows														
6. Desenvolvimento do protótipo do IDE proposto (parte visual)														
7. Integração do IDE com uma biblioteca de comando de voz														
8. Criação de uma interface para receber o comando de voz														
9. Testes de validação e Análise dos resultados obtidos														
10. Apresentação TCC I														
11. Novos testes														
12. Documentação														
13. Apresentação TCC II														

## 9 Referências

- SEBESTA, **Robert W. Conceitos de linguagem de programação**. 9.ed. Porto Alegre: Bookman, 2011
- MICROSOFT, 2015; **Site Oficial**. Disponível em < <https://msdn.microsoft.com/pt-br/>>. Acesso em 21 de Outubro de 2015.
- SILVA, Luciano; SILVEIRA, Ismar F. **Desenvolvimento de Aplicações em Realidade Aumentada e Diminuída para iPhone e iPad. Tendências e Técnicas em Realidade Virtual e Aumentada**. Porto Alegre. 2011. 1. p. 1-158. Disponível em < <http://comissoes.sbc.org.br/ce-rv/livro2011.pdf#page=129> >. Acesso em 11 de Novembro de 2015.
- NETBEANS, 2015; **Site Oficial**. Disponível em < [https://netbeans.org/index\\_pt\\_PT.html](https://netbeans.org/index_pt_PT.html)>. Acesso em 24 de Outubro de 2015.
- Desenvolvimento de um aplicativo para monitoramento de subestações de energia**. Seminário de Iniciação Científica. XXII. Ijuí. 2014. Disponível em < <https://www.revistas.unijui.edu.br/index.php/salaconhecimento/article/view/3799/3181> >. Acesso em 11 de Novembro de 2015.
- ANISZCZYK, Chris; GALLARDO, David. **Introdução à Plataforma Eclipse**. Disponível em < <http://www.ibm.com/developerworks/br/library/os-eclipse-platform/> >. Acesso em 11 de Novembro de 2015.
- PALMEIRA, Thiago Vinícius Varallo. **Conhecendo o Eclipse - Uma apresentação detalhada da IDE**. Disponível em < <http://www.devmedia.com.br/conhecendo-o-eclipse-uma-apresentacao-detalhada-da-ide/25589#ixzz3qdYRxJx0> >. Acesso em 11 de Novembro de 2015.

FERRADA, Romy Britt Hernandez. **Inclusão Digital de Sujeitos com Deficiência Física através do uso da Tecnologia Assistiva**. 2009. 152 f. Dissertação (Mestrado) - Curso de Educação, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2009. Disponível em: <<http://www.lume.ufrgs.br/bitstream/handle/10183/18387/000729561.pdf>>. Acesso em 11 de Novembro de 2015.

MICROSOFT. **Microsoft Speech**. Disponível em < <https://dev.windows.com/pt-br/speech> >. Acesso em 11 de Novembro de 2015.

MICROSOFT. **Microsoft Speech Platform**. Disponível em < [https://msdn.microsoft.com/en-us/library/hh361572\(v=office.14\).aspx#>](https://msdn.microsoft.com/en-us/library/hh361572(v=office.14).aspx#>). Acesso em 11 de Novembro de 2015.

ORACLE. **Java Speech API Frequently Asked Questions**, Disponível em < <http://www.oracle.com/technetwork/java/jsapifaq-135248.html#what> >. Acesso em 11 de Novembro de 2015.

ADORF, Julius. **Web Speech API**. Stockholm: Kth Royal Institute Of Technology, 2013. 11 p. Disponível em < <http://www.juliusadorf.com/pub/web-speech-api.pdf> >. Acesso em 11 de novembro de 2015.

ENDEN, Jarkko. **JAVA SPEECH API**. Finland: University Of Helsinki, 2001. 19 p. Disponível em <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.22.818&rep=rep1&type=pdf>>. Acesso em 11 de novembro de 2015.

SHIRES, Glen. **Voice Driven Web Apps: Introduction to the Web Speech API**. 2015. Disponível em <<https://developers.google.com/web/updates/2013/01/Voice-Driven-Web-Apps-Introduction-to-the-Web-Speech-API>>. Acesso em 11 de novembro de 2015.

REHOR, Kenneth G. **What is VoiceXML?** 2001. Disponível em <[http://www.voicexml.org/wp-content/uploads/sites/2/static/Review/Jan2001/features/Jan2001\\_what\\_is\\_voicexml.html](http://www.voicexml.org/wp-content/uploads/sites/2/static/Review/Jan2001/features/Jan2001_what_is_voicexml.html)>. Acesso de em 11 novembro de 2015.

GUILHOTO, Paulo José dos Santos; ROSA, Susana Patrícia Costa de Sousa. **Reconhecimento de Voz**. Coimbra: Universidade de Coimbra, 2001. Disponível em <<https://student.dei.uc.pt/~guilhoto/downloads/voz.pdf>>. Acesso em 11 de novembro de 2015.

SILVA, Ênio et al. **Desenvolvimento de um Sistema de Reconhecimento Automático de Voz Contínua com Grande Vocabulário para o Portugues Brasileiro**. Congresso da Sociedade Brasileira de Computação. XXV. São Leopoldo. UNISINOS. Disponível em < <http://www.nilc.icmc.usp.br/til/til2005/arq0069.pdf> >. Acesso em 11 de novembro de 2015.

APPLE. **Siri**. 2015. Disponível em: <<http://www.apple.com/br/ios/siri/>>. Acesso em 11 de novembro de 2015.

VOICENOTE. **VoiceNote II - Speech to text**. 2015. Disponível em: <<https://chrome.google.com/webstore/detail/voicenote-ii-speech-to-text/hfknjgplnkgjihghcidajefmldhibfm>>. Acesso em 11 de novembro de 2015.

SEESP/MEC. **Saberes e Práticas da Inclusão**. 2006. Disponível em: < <http://portal.mec.gov.br/seesp/arquivos/pdf/deficienciafisica.pdf> >. Acesso em 17 de novembro de 2015.

**Tecnologia Assistiva**. 2015. Disponível em: < <http://www.deficientesemacao.com/introducao> >. Acesso em 11 de novembro de 2015.

LEMONS, A. **Olhares sobre a cibercultura**. Porto Alegre: Sulina, 2003.