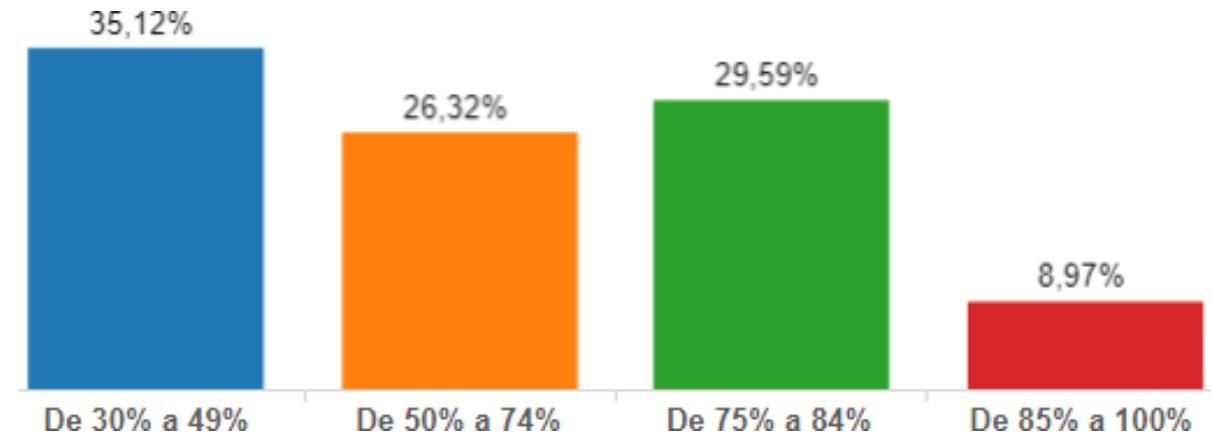


Construcción y Evolución de Software

SEMANA 5 – SESIÓN 1

Antecedentes

En el Ecuador, según el Consejo Nacional para la Igualdad de Discapacidades (CONADIS) existen 55.487 personas que padecen de discapacidad visual. El grado de discapacidad visual entre las personas no videntes se presenta en la siguiente figura, en donde se puede apreciar el 38.56% corresponde a personas con discapacidad visual grave y ceguera.



*Estadísticas hasta el año 2023

Descripción del Producto

Este producto de software permite transcribir textos a Braille y viceversa.

Finalidad

La finalidad del producto es que personas sin discapacidad puedan producir a bajo costo señalética o rotulación braille para mejorar la accesibilidad de edificios, aparatos, juegos de mesa, prendas de vestir, medicamentos, alimentos empacados, etc.

Objetivo del proyecto

Desarrollar un producto de software que genera textos en el sistema de lectoescritura braille en dos iteraciones, una por cada bimestre. Al culminar este proyecto, el estudiante tendrá las habilidades y conocimientos necesarios para leer y escribir documentos en formato Braille, además de una renovada comprensión sobre cómo contribuir a un mundo más inclusivo.

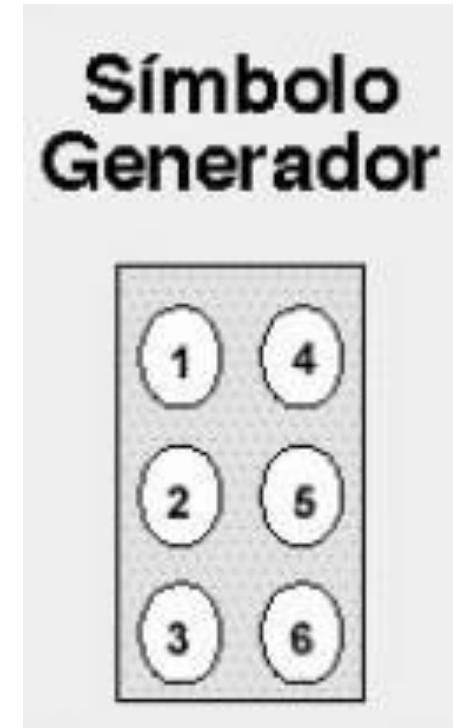
A continuación se detallan los requisitos para el primer bimestre:

Requisitos generales del cliente

El cliente requiere que el software permita:

- 1. Transcribir textos de español a braille incluyendo números, abecedario, vocales acentuadas, y signos básicos.**

El sistema parte del símbolo generador o cuadratín que es una figura rectangular, conformada por seis puntos en relieve dispuestos en dos columnas de tres puntos cada una. Cada punto del símbolo generador se identifica con un número diferente dependiendo de la posición espacial que ocupe en el rectángulo. El símbolo generador con indicación del número correspondiente a cada uno de sus seis puntos es el siguiente:



Requisitos generales del cliente

La primera serie matriz o primitiva, comprende las grafías correspondientes a las diez primeras letras del alfabeto (a-b-c-d-e-f-g-h-i-j).

a	b	c	d	e	f	g	h	i	j
•:	•:	••	••	••	••	••	••	••	••
1	12	14	145	15	124	1245	125	24	245

La segunda serie (k-l-m-n-o-p-q-r-s-t) resulta simplemente de añadir a la primera el punto 3.

k	l	m	n	o	p	q	r	s	t
•:	•:	••	••	••	••	••	••	••	••
13	123	134	1345	135	1234	12345	1235	234	2345

Requisitos generales del cliente

La tercera serie (u, v, x, y, z) es el resultado de añadir a la primera serie los puntos 3 y 6.

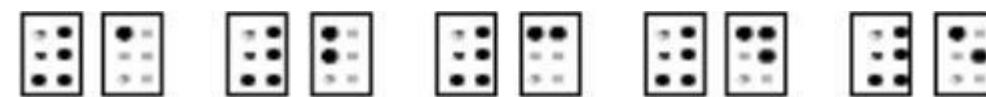
u	v	x	y	z
136	1236	1346	13456	1356

Letras adicionales:

ñ	w	á	é	í	ó	ú	ü
12456	2456	12356	2346	34	346	23456	1256

Requisitos generales del cliente

Escritura de números. Los diez dígitos se obtienen anteponiendo el signo de número a la primera serie:



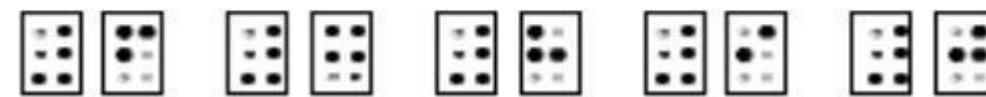
1

2

3

4

5



6

7

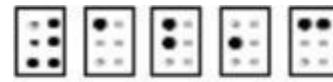
8

9

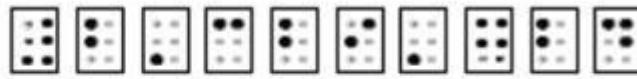
0

Requisitos generales del cliente

Para cantidades de dos o más cifras, se coloca el signo de número solamente al principio, también se puede colocar puntos y comas. Los números se separan con espacios en blanco.



12,3



2.329.724

Cuadro de resumen

LibroCiego	1	4	SISTEMA DE LECTO-ESCRITURA. BRAILLE ESPAÑOL		
2	5		Cuadratín	Mayúsculas	Números
3	6				
ABECEDARIO					
	a b c d e f g h i j				SIGLOS
	k l m n o p q r s t				
	u v w x y z	ñ ü	()	+ x = ÷ -	
	á é í ó ú				NÚMEROS
			1 2 3 4 5		
			6 7 8 9 0		

Requisitos generales del cliente

2. Generar señalética braille a partir de textos en español.

Ejemplo de impresión de señalética para ascensores (con aplicación Windows):



Texto en tinta:

0 1 2 3 4 5 6 7

Texto en braille:

JM J' JM J'' JM J''' JM J'''' JM

Considerar

La aplicación puede ser de escritorio, web, móvil. Cada equipo debe presentar en un repositorio en github:

Rama main el código en la versión estable.

Rama documentacion incluir lo siguiente

1. Diseño arquitectónico de alto nivel
2. Documentación del ambiente de desarrollo: herramientas seleccionadas y flujo de trabajo (no se puede desarrollar en la rama main, consultar una estrategia de ramificación, documentarla y aplicarla)
3. Documentación técnica sobre el código fuente por ejemplo JavaDoc.
4. Documentación de casos de prueba y resultados de la ejecución de cada caso. Si el caso de prueba no fue exitoso al inicio es obligatorio incluir un análisis, la solución y ejecutar nuevamente el caso de prueba.
5. Manual de instalación
6. Manual de usuario