

Sensores 2019. Igaciencia



Visor estereoscópico de gráficos vectoriais interactivos

Alumnado:

Ariza Abaña, Alejandro

Gil Seoane, Miguel

Hervella Sáinz de la Torre, Julia

Patrón Mayo, Álvaro

Rudiño Lorenzo, Breixo

Ruiz Abete, Carlos Andrés

Sixto Poulsen, Víctor Sloth

Suárez Montes, Óscar

Veiga Feteira, Iván

Villar Abeijón, Javier

Curso: **1º Bacharelato** Grupo: **A**

Materias: **Debuxo Técnico I e Tecnoloxía industrial I**

Profesorado: **Luz Ramos López e Xabier Lorenzo Abalde**

IES de Brión

Visor estereoscópico de gráficos vectoriais interactivos

Índice:

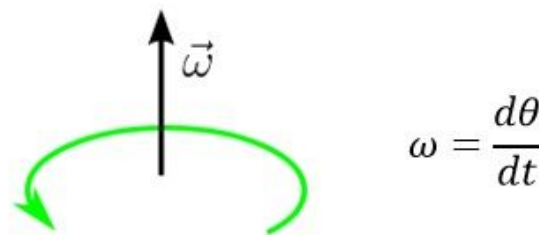
1. Introducción	2
2. Obxectivos	3
3. Materiais, programas e recursos . Gráfico da montaxe	4
4. Desenvolvemento do proceso	5
5. Dificultades	10
6. Análise de resultados. Conclusións	10
7. O número de cuestións ás que responde e a súa calidade	11
8. Programación efectuada	11
9. Referencias	12

1. Introducción

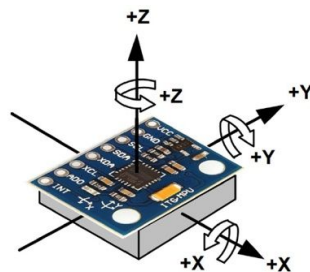
Os dispositivos móbiles actuais están dotados de sensores, que nos permiten interactuar cos mesmos. Neste proxecto desenvolvemos o deseño dun visor de gráficos interactivos con html5 e JavaScript que nos permiten o uso de dispositivos móbiles para obter experiencias inmersivas de realidade virtual por medio destes sensores.

Os principais sensores empregados no proxecto son:

- Acelerómetro: Dispositivo empregado para que o móbil sepa en qué orientación está colocado, de maneira que o dispositivo sepa cando o estás mirando en horizontal, vertical ou boca abaixo. O concepto empregado polos acelerómetros para medir a aceleración é $a=F/m$, tendo en conta que a pesar de que non exista movemento, o acelerómetro sempre está medindo a aceleración da gravidade.
- Xiroscopio: Sensor que mide a aceleración non gravitacional e que cumprimenta a información sobre a orientación do móbil que ofrece o acelerómetro aportando a medida da rotación e xiro do móbil. Co xiroscopio podemos medir a velocidade angular, a tasa de cambio do desprazamento angular por unidade de tempo.

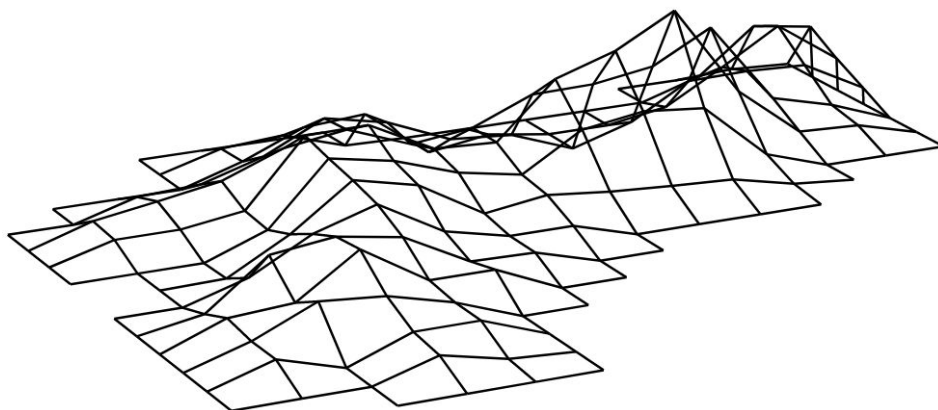


Realmente cando falamos de xiroscopio estamos referíndonos a unha combinación de acelerómetro, xiroscopio e brúxula. Estes proporcionannos os datos *alfa*, *beta* e *gamma*. A súa vez, estes datos poden ser recollidos por diversos programas e por eventos como o `deviceOrientation`, que neste caso empregamos para o noso proxecto.



2. Obxectivos

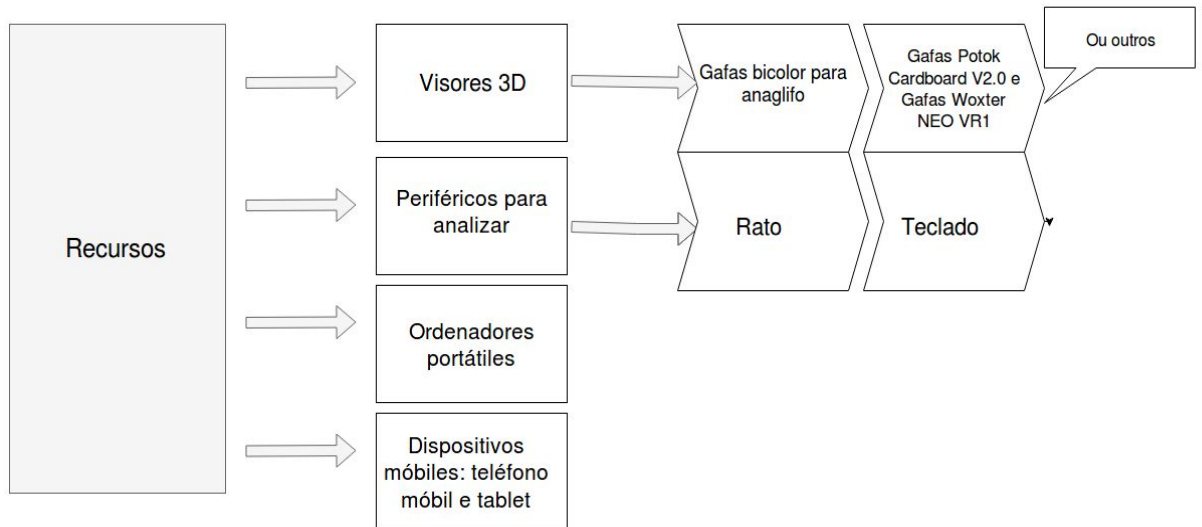
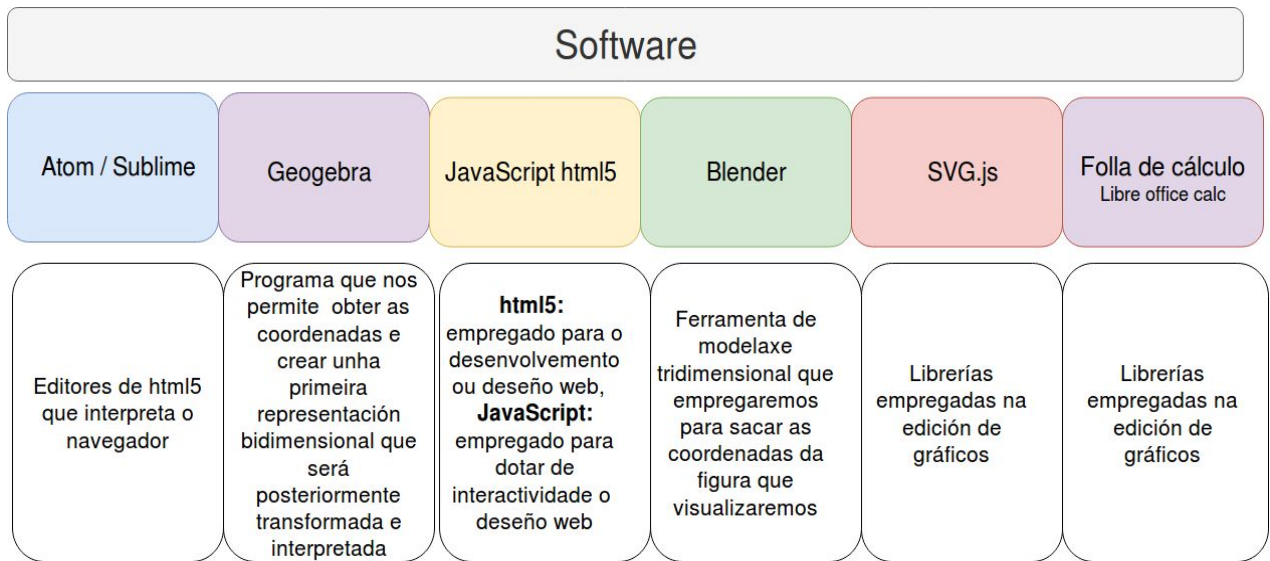
O obxectivo principal é a creación dun visor de obxectos, espazos ou lugares que nos permita obter unha experiencia inmersiva de Realidade Virtual (RV) empregando o xiroscopio dun dispositivo móbil. No noso traballo en particular aplicámolo para a visualización da Illa de San Martiño, illa sur do arquipélago das Illas Cíes.



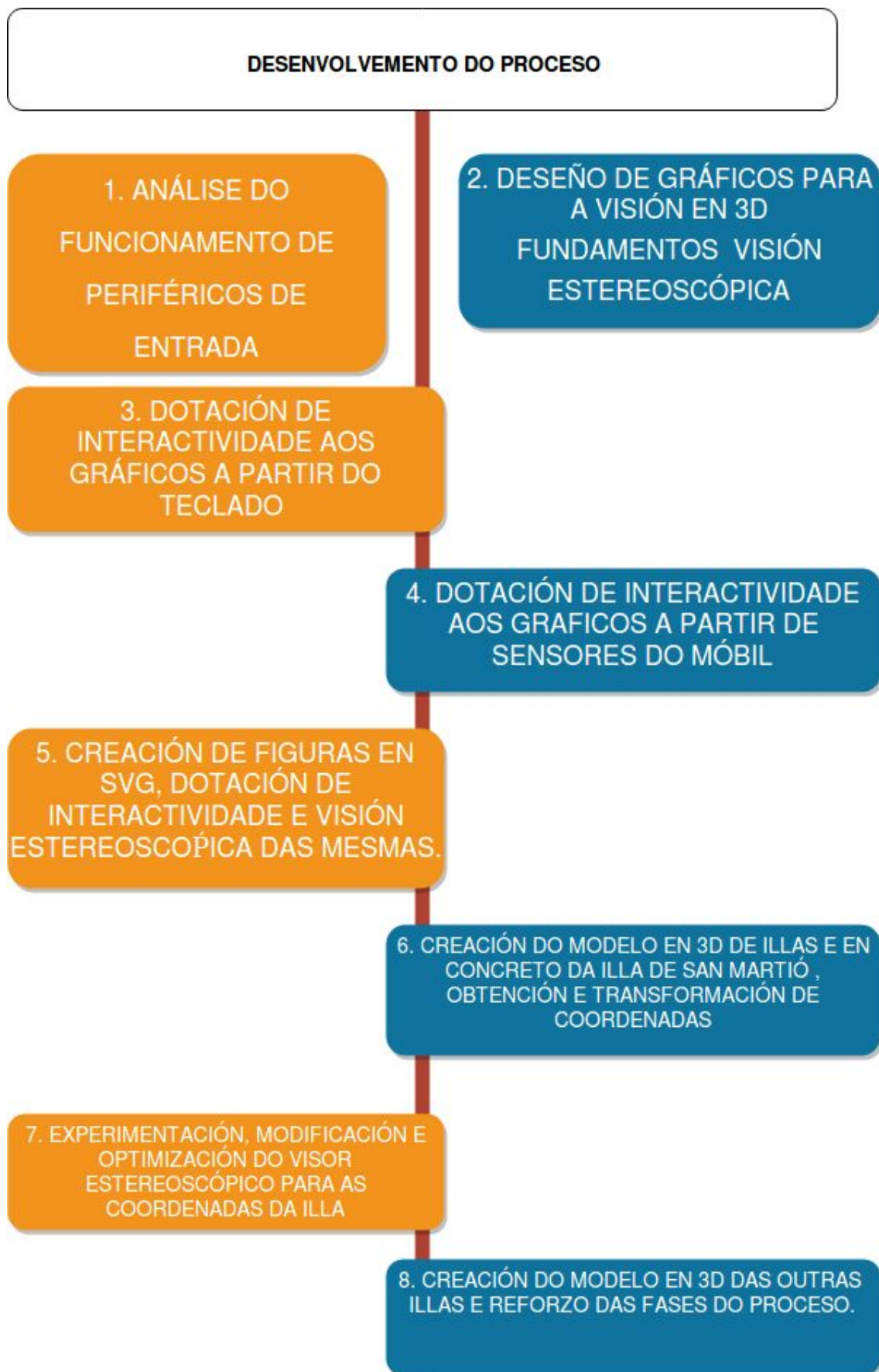
Para acadar o obxectivo final, establecemos diferentes obxectivos parciais:

- Experimentar co comportamento de periféricos de entrada de información.
- Crear gráficos que nos permitan comprobar e analizar os fundamentos da estereoscopia.
- Enviar información en forma de movementos básicos a gráficos simples coas frechas do teclado empregando html5 e JavaScript.
- Experimentar co uso dos esvaradores nos applets de Geogebra para realizar gráficos interactivos.
- Experimentar con prototipos realizados en geogebra e svg como diferentes opcións para o deseño de gráficos interactivos.
- Interaccionar cos gráficos creados a partir do xiroscopio do dispositivo móbil, empregando as gafas de realidade virtual.
- Crear e optimizar o visor empregando html5 e JavaScript.
- Crear librerías de figuras con gráficos svg empregando svg.js.
- Modelar en Blender figuras a partir de mapas cartográficos en diferentes formatos.
- Obter as coordenadas de modelos tridimensionais.

3. Materiais, programas e recursos . Gráfico da montaxe



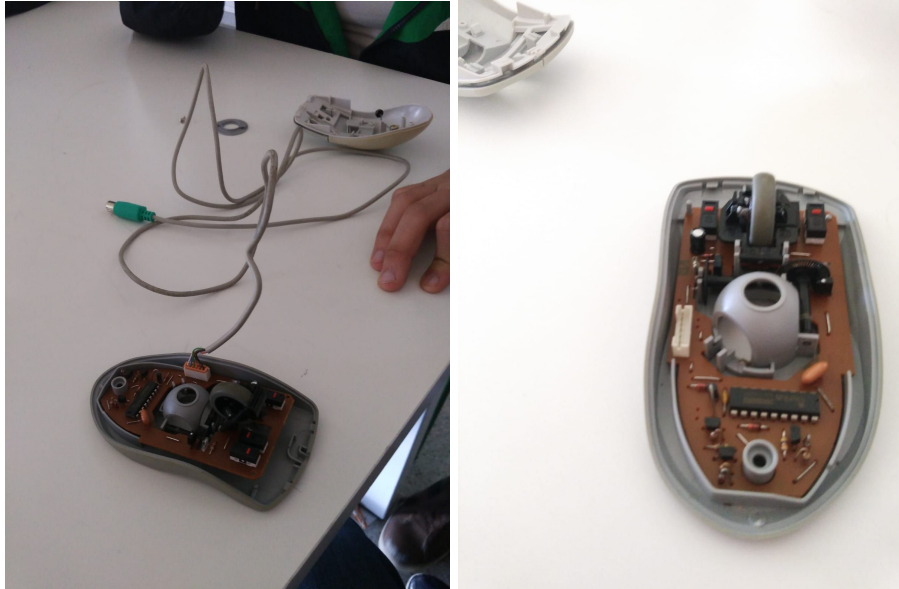
4. Desenvolvemento do proceso



FASE 1: Análise do funcionamento de periféricos de entrada

Sesións introdutorias orientadas a investigar os dispositivos básicos de entrada de datos.

Análise do funcionamento de diferentes periféricos (rato e un teclado) para entender e comprobar o seu funcionamento, como se despraza o cursor na pantalla e que posibilidades de interacción nos permiten.

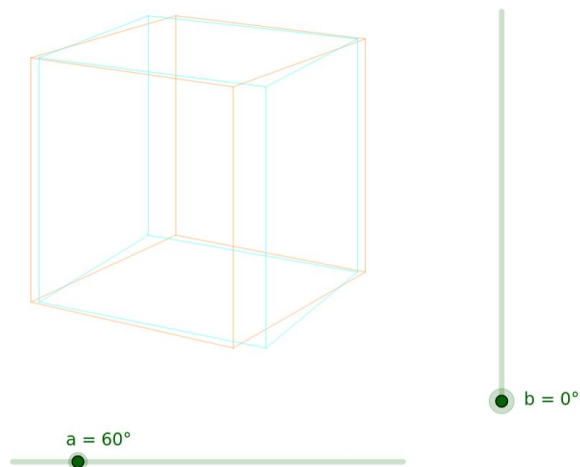


FASE 2: Deseño de gráficos para a visión en 3D

Estudamos os fundamentos da visión estereoscópica.

Construímos en Geogebra unha ferramenta para representar anaglifos a partir das proxeccións horizontal e vertical dos vértices de calquera figura.

Debuxamos en Geogebra diferentes figuras parametrizadas e reguladas con esvaradores. Experimentamos cos resultados.



Definimos, a partir de modelos en diédrico, as relacións trigonométricas para poder xirar puntos dadas as súas coordenadas en torno aos eixes z e x.

$$\begin{aligned}xP_1 &= \cos(\alpha) * x - \sin(\alpha) * y \\yP_1 &= (-\sin(\alpha) * x - \cos(\alpha) * y) * \cos(\beta) - \sin(\beta) * z \\yP_2 &= (-\sin(\alpha) * x - \cos(\alpha) * y) * \sin(\beta) + \cos(\beta) * z\end{aligned}$$

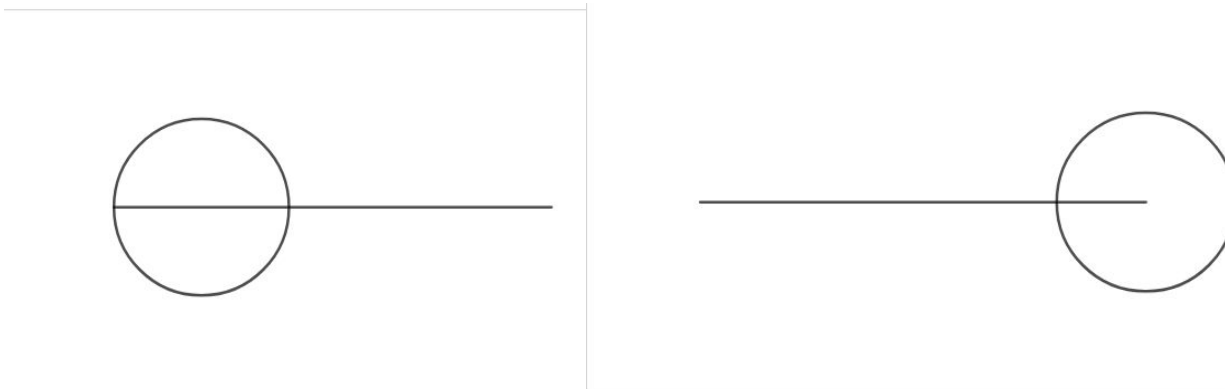
Definimos tamén as expresións que nos permiten obter as proxeccións cónicas dun punto a través das ecuacións das proxeccións vertical e horizontal do raio proxectante.

$$\begin{aligned}xP' &= -yP_1 * (xV_1 - xP_1) / (yV_1 - yP_1) + xP_1 \\yP' &= (-yP_1 * (xV_1 - xP_1) / (yV_1 - yP_1)) * (yV_2 - yP_2) / (xV_1 - xP_1) + yP_2\end{aligned}$$

FASE 3: Dotación de interactividade aos gráficos co teclado do ordenador

Deseñamos unha páxina en html5 que incorpora un applet de Geogebra e un script de JavaScript que dota de interactividade a figura usando as flechas de desprazamento dun teclado. A figura creada en GeoGebra está regulada por esvaradores e estes, a súa vez, son modificados dende o script cun métodos específicos de Geogebra para JS. Así a figura pode moverse ao longo dun segmento, tanto cara á dereita como á esquerda.

[Descarga o arquivo](#)



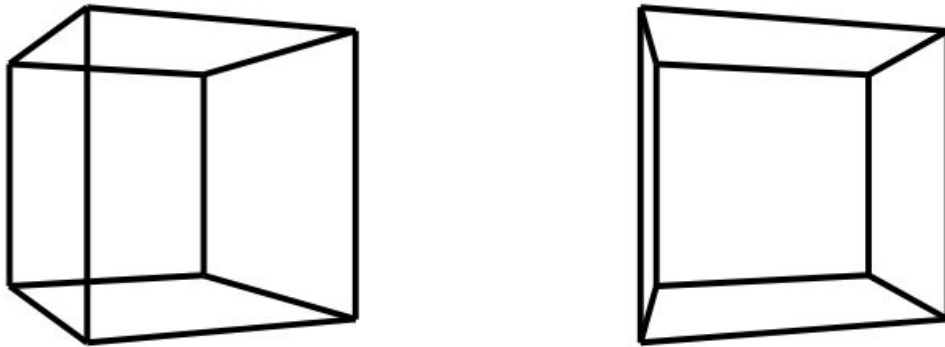
FASE 4: Dotación de interactividade aos gráficos a partir do sensores do móbil, neste caso a partir do xiroscopio.

Modificación do programa, substituíndo o evento de teclado pola toma de datos dos sensores do teléfono empregando o evento **deviceOrientation**. Coa simple orientación do noso teléfono ou dispositivo móbil, podemos interactuar coas figuras recollendo uns valores angulares (alpha e gamma) que o móbil recolle mediante o xiroscopio. Para poder empregar estes datos coa librería **svg.js** temos que pasalos de graos a radiáns, para traballar con applets de Geogebra non. Para experimentar o funcionamento creamos o programa agullas.

[Descarga o arquivo](#)

FASE 5: Creación de figuras en svg, dotación de interactividade e visión estereoscópica das mesmas.

A partir das expresións calculadas na fase 2 e empregando a librería svg.js, programamos en JavaScript unhas funcións que permiten transformar unha nube de puntos definida polas súas coordenadas en dúas imaxes en perspectiva da figura que representa (unha para cada ollo).



Representamos un cubo e varios poliedros simples en svg e a partir das súas coordenadas.

Para clarificar o traballo decidimos diferenciar o código do visor e o código de cada unha das figuras representadas, polo que definimos as figuras nun arquivo js aparte. No arquivo principal establecemos as relacións entre esas coordenadas orixinais os ángulos de xiro e os puntos de vista.

Para experimentar o funcionamento descarga o programa.

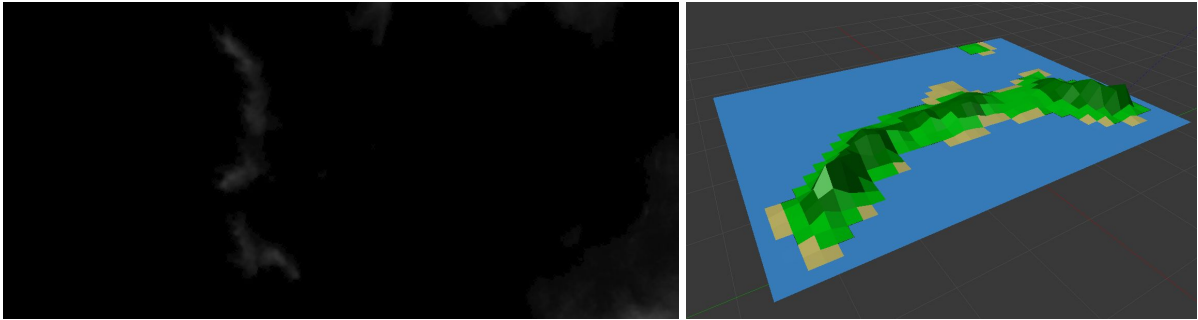
[Descarga o arquivo](#)

FASE 6: Creación do modelos en 3D de illas e en concreto da Illa de San Martiño , obtención e transformación das coordenadas.

Esta fase do traballo consistiu en transformar un mapa topográfico da Illa Sur das Cies ou Illa de San Martiño nun modelo 3D poligonal, e máis concretamente en definir as coordenadas e organizar a nube de puntos que o configuran.

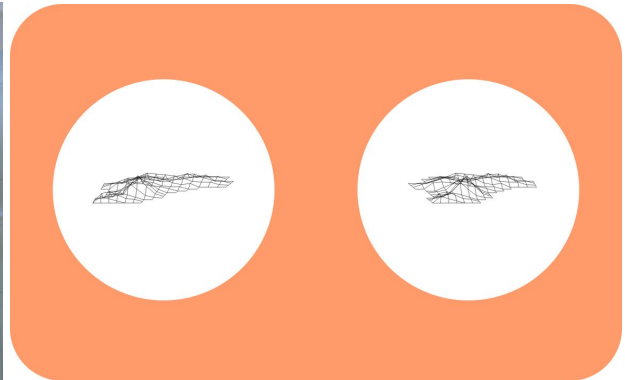
Para iso obtivemos un mapa de alturas da web [terrain.party](#). Unha imaxe png que contén a información das cotas do relevo nos diferentes tonos de gris empregados (a maior altura máis claro).

A continuación, en Blender, modificamos unha retícula ortogonal coa textura do mapa, obtendo así unha malla, ou nube de puntos que definen a orografía da illa.



Introducimos nunha folla de cálculo as coordenadas de todos os puntos da malla, facendo coincidir as celdas, coa posición de cada punto na planta da illa. Deste modo as filas e columnas de datos corresponderíanse directamente coas liñas poligonais que posteriormente pretendíamos trazar no svg.

A partir dos datos desta folla, e xa en JavaScript, definimos uns arrays coas coordenadas de cada unha das liñas poligonais para podelas representar no visor.



FASE 7: Experimentación, modificación e optimización do visor estereoscópico para as coordenadas da Illa.

Realizamos probas do visor con diferentes dispositivos móbiles e navegadores, para observar posibles melloras, localizar erros e a realizar modificacións na programación.

Os principais foron:

- Axustar o grosor das liñas.
- Axustar tamaño e centrar os trazados.
- Diseñar dous botóns para regular a distancia interpupilar.
- Limitar a frecuencia de refresco dos gráficos mediante unha animación para non saturar os dispositivos e navegadores.

Para experimentar, descarga o programa Illa_sur nun dispositivo móbil.

[Descarga o arquivo](#)

FASE 8: Creación do modelo 3D das restantes illas.

Esta fase está en progreso na actualidade e servirá para afianzar o proceso desenvolvido ata o momento, permitindo a todo o alumnado do grupo experimentar as diferentes fases e reforzar os coñecementos traballados.



5. Dificultades

- Introducir, comprender e familiarizarnos con novas linguaxes de programación e/ou aplicacións de desenvolvemento web html5, JavaScript e representación de gráficos e modelado svg.
- Realizar as transformacións de coordenadas dun obxecto tridimensional para conseguir rotalo e proxectalo sobre un plano.
- Marcar todos os puntos de coordenadas do mapa 3D.
- Tratamento do alto número de coordenadas obtidas.
- Falta de sistematización na xestión da documentación cos logros conseguidos.

6. Análise de resultados. Conclusións

O resultado principal é a elaboración dun visor de figuras virtuais que nos permite obter unha experiencia de RV inmersiva empregando o xiroscopio (sensor do móbil) como dispositivo de entrada do movemento, html5, JavaScript como aplicacións

para o visor e a librería svg.js. Aplicamos o visor para obter a experiencia de realidade virtual sobre a Illa de San Martiño, ou Illa Sur das Cíes.

O visor pode empregarse para calquera figura, espazo ou obxecto a partir das súas coordenadas, o que permite múltiples aplicacións e obter a experiencia de RVI.

Antes de chegar ao resultado final, fomos acadando os seguintes resultados parciais:

- Representar figuras en geogebra e svg que nos permiten experimentar coa visión estereoscópica.
- Elaborar programas que permiten desprazar un obxecto a partir do teclado.
- Elaborar programa que permiten desprazar un esvarador a partir do movemento do móbil, incorporando o xiroscopio, como sensor.
- Obter as coordenadas dunha figura a partir dun modelo 3D ou nube de puntos.

7. O número de cuestións ás que responde e a súa calidade

Economía de medios. Só é preciso dispor dun dispositivo móbil con xiroscopio, un navegador que soporte deviceOrientation e un visor binocular de RV que pode ser de cartón.

A experiencia inmersiva é realmente intensa no referente á percepción tridimensional das figuras debido ao deseño estereoscópico dos gráficos.

O proxecto é escalable coa posibilidade de aplicación do visor a outros obxectos, figuras e lugares.

Os resultados poden ter moi diversas aplicacións incluíndo a posibilidade de introducir tamén realidade aumentada sobre un entorno real.

As aplicacións empregadas para realizar o proxecto son todas de software libre.

O aproveitamento de recursos e datos liberados por outros proxectos.

8. Programación efectuada

Programación:

- [Visor estereoscópico](#)
- [Librería da illa](#) San Martiño do parque natural Illas Cíes
- [Librería SVG](#)

Aclaracións do programa:

- **Variables:**

- Primeiro definimos variables, entre outras un espazo para inxectar o noso svg, as coordenadas naturais dun punto xenérico, as súas coordenadas rotadas, as proxectadas, ángulos de xiro, factores de desprazamento, etc.

- Definimos tamén posición do punto de vista estereoscópico, o que fai que teñamos dous puntos de vista separados por unha distancia. Esta distancia está definida nunha variable chamada desprazamento, que foi previamente calculada en función

da resolución da pantalla do dispositivo e que pode modificarse mediante botóns en pantalla.

- **Chamamos a dúas librerías:**

-SVG(<http://svgjs.com>): Librería ben documentada na súa páxina, sérvenos para facilitar o trazado de gráficos vectoriais.

-Figura: Esta librería recolle as coordenadas das arestas da figura antes de ser transformadas.

- **Creación de funcións gráficas:**

-Crear SVG: creamos o contexto no que a figura será representada

-Función transformar punto: teñen lugar dúas transformacións; unha de rotación dos puntos segundo dous ángulos, alpha e gamma, que proveñen do deviceOrientation e outra de proxección cónica con dous puntos de vista.

-Función de transformar segmento: executa dúas veces e para os extremos dun segmento, a función de transformar punto, acumulando os valores dos resultados para conseguir representar unha aresta.

-Debuxar: Traza a aresta calculada.

-Debuxar Figura: borra os trazados anteriores, crea un novo svg e refresca o trazado de todas as arestas da figura.

- **Device Orientation:**

Mediante a función obterOrientacion e co método deviceOrientation recollemos os valores, en grados, de alpha e gamma servidos polo dispositivo e convertémolos a radiáns para poder mover a figura co movemento do dispositivo.

9. Referencias

Acelerómetros

- [Artigo de xataka sobre o xiroscopio nos móbiles](#)
- [Outros sensores no móbil](#)
- [Xiroscopio en proxectos arduino](#)

Realidade Virtual

- [Realidade virtual en Jscon WebVR](#)
- [Realidade virtual](#)
- [Xogo con Realidade Virtual](#)
- [Crea a túa primeira aplicación de Realidade Virtual con A-Frame](#)
- [10 demos e experimentos webVR](#)

Terrain Party

- [Aplicación en liña para obter mapas de altura.](#)

Tutoriais de JavaScript

- En W3Schools.com [Titorial JS](#)
- En DesarrolloWeb [Titorial JS](#)

Métodos de Geogebra para JavaScript:

- <https://wiki.geogebra.org/es/Referencia:JavaScript>

Perspectiva cónica con Ggb.

- [Perspectiva cónica 360X360](#)
- [Perspectiva cónica dun punto. Determinación de coordenadas](#)
- [Perspectiva cónica dun punto. Macro](#)
- [Anaglifo de cubo 360X360](#)

Axonometría con Geogebra

- [Recurso de introdución ao sistema axonométrico de representación](#)
- [Axonometría de punto. Proxección directa](#)
- [Interprete de geogebra](#)

[Referencias básicas para traballar con Applets de GGB e Device Orientation](#)
[Librería svg](#)