



CodeCraft Hacklab

El reto de enseñar programación a niños
Un proyecto de divulgación tecnológica
para la infancia y la juventud

Cátedra de Telefónica 2014

Contenido

Descripción y objetivos del proyecto	1
La hora del código	2
Scratch Day	3
Contexto	4
Antecedentes y estado actual de la materia	4
Experiencia del equipo de trabajo en la materia	5
Bibliografía.....	5
Referencias en la red.....	5
Justificación y viabilidad	7
Beneficios previstos del proyecto	7
Encaje en las áreas de interés de la Cátedra.....	7
Impacto previsto en la comunidad universitaria	8
Viabilidad funcional y económica	8
Plan de trabajo	10
Metodología docente	10
Metodología del proyecto	10
Equipo humano de trabajo y funciones asignadas	11
Medios materiales.....	12
Calendario e hitos principales	11
Resultados del proyecto.....	13
Resultados esperados.....	13
Plan de difusión y diseminación de resultados.....	13
Presupuesto económico	14

Descripción y objetivos del proyecto

Saber programar representa una llave que abrirá muchas puertas a los actuales estudiantes. Sin embargo en casi ninguna escuela o colegio se les enseña a los niños a programar

Para hacer frente a esta situación, en Estados Unidos ha surgido una iniciativa conocida como “Hour of Code”, donde se introduce a los estudiantes en el mundo de la programación de manera lúdica (www.code.org).

Muchas figuras públicas como Bill Gates, Shakira, Ashton Kutcher o Mark Zuckerberg han participado en esta iniciativa en la que animan a los estudiantes a aprender a programar. Así mismo, empresas como Google, Yahoo, Dropbox, Facebook, Microsoft y Apple contribuyen económicamente a esta iniciativa.

Desde la Universidad de Las Palmas, podemos y debemos promover esta iniciativa en nuestro entorno, fomentando el aprendizaje de la programación entre los estudiantes de primaria y secundaria.

En este caso, la idea consiste en animar a los niños a entrar en el mundo de la programación. Programar puede ser tan emocionante como jugar a un video juego. Se trata de que los niños se diviertan y al mismo tiempo aprendan una habilidad que les servirá de por vida.

Para ello se plantea la creación de un hacklab, un colectivo autogestionado con el interés en la vertiente social de la tecnología: uso de software libre y enseñar a los niños y jóvenes a programar.

La idea de los hacklabs viene de Italia, donde desde 1998 se están creando diversos colectivos relacionados con la tecnología para desarrollar actividades como la formación técnica, la difusión del software libre, talleres prácticos, el refuerzo de las habilidades comunicativas y emprendedoras. En España hay diversos hacklabs creados por toda la geografía, en sitios como Zaragoza, Bilbao, Alicante, Navarra, Madrid...

En el CodeCraft Hacklab se plantean inicialmente dos ejes de actuación:

1. Ofrecer a los colegios una actividad extraescolar complementaria llamada “La hora del código”.
2. Organizar un evento Scratch Day en la Universidad de Las Palmas

La hora del código

Esta actividad se ofrecería desde la Universidad a los colegios de forma que la pudieran incluir en su proyecto educativo. Sería una actividad que se realizaría fuera del entorno escolar, al igual que se realizan otras actividades como visitas a museos, fábricas o excursiones educativas, y serviría para la enseñanza-aprendizaje, de una forma más creativa y práctica, diferente a lo rutinario.

La actividad se ofrecería a niños entre 8 y 14 años. En total, se organizarán 20 talleres distribuidos a lo largo del año, a los que se invitará a los colegios a asistir. Para ello, los colegios se podrán inscribir a través de la página web del proyecto.

El tema del taller será Minecraft que se incorpora como elemento vertebrador de la actividad. Minecraft es un videojuego actualmente muy de moda entre los niños y que sirve como elemento de atracción para introducirlos en el mundo de la programación. En Minecraft no hay un objetivo específico y el jugador tiene libertad para desplazarse por un mundo virtual generado automáticamente. Este mundo está formado por bloques tridimensionales que representan madera, piedra, tierra, agua, nieve, lava, arena, hierro, diamante, carbón... Los jugadores intervienen a través de un avatar y el juego está centrado en la recolección y colocación de estos bloques para realizar construcciones como una casa, un castillo, un poblado...



Figura 1. Minecraft

El juego ofrece la posibilidad de ser intervenido a través de un programa con el que se pueden automatizar la creación de estas construcciones. Realizando pequeños algoritmos, los niños podrían realizar órdenes para automáticamente construir una pared, cavar un foso, realizar una escalera...

Minecraft se puede programar con lenguajes como Javascript o Python, que son sencillos de utilizar. No obstante, se deben gestionar los aspectos motivacionales para evitar la frustración por la dificultad que la propia actividad tiene. En este sentido, Minecraft es el elemento clave para atraer y mantener la atención de los niños: a través de la programación el niño tiene la posibilidad de poder controlar un avatar, introducir logros de complejidad creciente, y a la vez poder establecer relaciones con otros niños con los que compartir intereses comunes.

Scratch Day

Scratch es un entorno de aprendizaje de lenguaje de programación. Está orientado fundamentalmente a ser usado por los niños, y ha sido desarrollado en el Media Lab del MIT (Instituto Tecnológico de Massachusetts).

En Scratch, el usuario combina diferentes tipos de componentes para realizar historias interactivas, juegos y pequeñas animaciones. Este entorno ayuda a pensar de forma algorítmica y a aprender a resolver problemas de forma metódica.



Figura 2. Scratch

El Scratch Day es un evento mundial, impulsado por el MIT, tiene como objetivo atraer el interés de los niños a la programación con Scratch, este año se celebrará el 14 de mayo. El año pasado participaron 188 entidades de 47 países de todo el mundo, siendo una oportunidad para poner en valor el compromiso de servicio de la ULPGC a la sociedad.

Contexto

Antecedentes y estado actual de la materia

Cuando un niño deja de escuchar explicaciones y en lugar de ello programa, el proceso de aprendizaje se transforma. El construccionismo (Papert, 1980) plantea aprovechar las TICs en la educación, y existen fabulosas oportunidades actualmente para hacerlo.

Según esta teoría del aprendizaje, el conocimiento se construye mediante la integración progresiva de las acciones. Esto sucede sobre todo en un contexto en el que el alumno está dedicado conscientemente en la construcción de una entidad que otros pueden observar, ya sea un castillo de arena en la playa o un programa de ordenador. El aprendizaje se produce a través de decisiones en vez de potenciales cognitivos, es decir, las ideas se forman y se transforman cuando se expresan en determinados contextos. Expresar ideas las hace tangibles y compartibles y a su vez nos ayuda a comunicarnos con los demás a través de nuestras expresiones. El ciclo de aprendizaje es un proceso iterativo mediante el cual los alumnos inventan por sí mismos las herramientas que mejor apoyan la exploración de lo que más les importa.

Para dar soporte a esta filosofía educativa, se han desarrollado varios entornos de programación que insisten en la actividad manipulativa del niño. Entre los entornos más importantes se encuentran:

1. Logo. Fue el primer lenguaje que se creó con una finalidad específicamente pedagógica por los propios investigadores que propusieron el construccionismo como teoría del aprendizaje (Seymour Papert)
2. Scratch. Es un lenguaje de programación orientado específicamente a niños y adolescentes con un concepto muy didáctico basado en la utilización de bloques que se unen para crear historias interactivas, animaciones, juegos, piezas musicales.
3. KPL (Acrónimo de Kids Programming Language) es una adaptación de Visual Basic .NET. Es un poco más avanzado que los anteriores en términos de herramientas que permiten desarrollar programas y juegos muy visuales y potentes.
4. Alice. Es un entorno de desarrollo que permite crear películas animadas o videojuegos sencillos, a través del control de objetos y personajes 3D en un mundo virtual.
5. Minecraft es un videojuego de construcción, de tipo «mundo abierto» en el que los jugadores pueden realizar construcciones mediante bloques tridimensionales que representan distintos

elementos de la naturaleza, como tierra, piedra, minerales o madera. El juego permite que se pueda programar automáticamente la construcción del mundo.

Experiencia del equipo de trabajo en la materia

El equipo con el que se cuenta para la ejecución del proyecto está formado por docentes e ingenieros informáticos. El equipo está conformado por muchos jóvenes ingenieros (menos de 26 años) no vinculados contractualmente con la Universidad pero con la ilusión de participar en una iniciativa que en otras muchas ciudades de España y el mundo se está realizando. La idea justamente es la de coparticipación y configurar un espacio abierto en el que puede integrarse quien desee contribuir en los valores promovidos en el Hacklab.

Además contamos con el asesoramiento y colaboración activistas de Agile Spain (comunidad de aplicación de metodologías ágiles) que coordinan otros hacklabs en España: **Zagales hacklab** (<http://zagales-hacklab.org/>) y **Unutopic Hacklab**.

La colaboración con estos hacklabs, que ya tienen la experiencia de haber trabajado en la enseñanza de la programación con niños y con métodos ágiles, contribuirá a traer a Canarias buenas prácticas y conocimiento de otros sitios de España.

Bibliografía

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc..

Harel, I., & Papert, S. (1990). *Software design as a learning environment*. *Interactive learning environments*, 1(1), 1-32.

Pólya, George (1965). *Cómo Plantear y Resolver Problemas*. Editorial Trillas. ISBN 968-24-0064-3.

Referencias en la red

<http://www.code.org>

<http://scratch.mit.edu>

<http://day.scratch.mit.edu>

http://info.scratch.mit.edu/es/Support/Scratch_Cards

<https://minecraft.net/>

<http://www.theguardian.com/education/series/digital-literacy-campaign>

<http://www.theguardian.com/teacher-network/2012/jan/10/how-to-teach-code>

<http://www.codecademy.com/>

Justificación y viabilidad

Beneficios previstos del proyecto

Programar implica estructurar el pensamiento, las ideas, convertirlas en un proyecto de construcción para generar algo nuevo. Programar también ayuda a desarrollar la creatividad y desarrollar la expresión artística y plástica. Cuando la programación se realiza en pareja (pair programming) se favorece mucho el trabajo en equipo.

Desde esta perspectiva, el objetivo de este proyecto es favorecer en los niños el desarrollo del pensamiento abstracto y el pensamiento algorítmico. Además, ayuda a poner en marcha procesos creativos que pueden ser realizados de forma cooperativa. También permite combinar diferentes capacidades como la lingüística, matemática, artística, espacial, interpersonal e intrapersonal.

Programar es una actividad que debe realizarse con una visión internacional, por tanto además contribuirá a fortalecer en los niños y padres el valor que tiene aprender el inglés, como vehículos de comunicación y desarrollo profesional.

En definitiva, aprender a programar para estos niños tiene un beneficio directo en la adquisición de la competencia digital, pero también al permitirle adquirir capacidades para abordar la resolución de problemas y el trabajo en equipo.

Por otro lado, con esta iniciativa se puede constituir en Canarias un hacklab permanente, como otros muchos que existen ya en el resto de España, aprovechando además la experiencia de otros hacklabs y el uso de métodos ágiles.

Encaje en las áreas de interés de la Cátedra

En este proyecto se plantea un uso innovador de las TIC para el gran público y en concreto estaría dirigido a la juventud y la infancia.

Tendría una gran visibilidad, tanto la actividad extraescolar como el Scratch Day, a la vez que una oportunidad para potenciar las competencias digitales de los niños y otras capacidades intelectuales como la resolución de problemas y el trabajo en equipo.

El sentido de incorporar la programación es cultural, social, pedagógico y tecnológico. Enseñar a programar, como antes aprender a leer y escribir, debe ser uno de los objetivos esenciales de la educación obligatoria. Saber leer y escribir constituye el vehículo por excelencia a través del cual las personas acceden al conocimiento cultural en nuestras sociedades. Desde esta idea surgió en el siglo XIX el afán de alfabetización de toda la población que actuara como herramienta de igualdad entre las personas y como instrumento de compensación de las desigualdades sociales. Hoy sabemos que el acceso al conocimiento es algo mucho más amplio, y en este sentido, el aprendizaje de la programación es mucho más que el conocimiento de la tecnología. Implica una forma razonar y enfrentarse a la información que son cruciales para el futuro de los niños y los jóvenes. Nuestro mundo actual es muy diferente al de hace 20 años y los niños deben también aprender a saber seleccionar y gestionar el exceso de información.

El proyecto contribuirá también a la formación de un equipo ágiles coparticipando en iniciativas relacionadas con la enseñanza de la programación proyectando hacia la sociedad conocimiento en tecnologías y metodologías ágiles.

Impacto previsto en la comunidad universitaria

En la práctica es imperativo que la comunidad universitaria proyecte valores de compromiso, responsabilidad y respeto hacia la sociedad. Las instituciones educativas debemos ser un ejemplo de eficiencia en términos económicos y generosidad en el ámbito social. Este proyecto permite a la ULPGC mostrar su implicación en el proceso de transformación social y educativa.

Viabilidad funcional y económica

Para la ejecución de este proyecto, tenemos disponibilidad de

1. Aulas, en la Escuela de Ingeniería Informática
2. Recursos materiales: ordenadores, proyectores, pizarras electrónicas
3. Recursos software: sistemas operativos, entornos de programación...

Necesitamos inicialmente una pequeña inyección de dinero para cubrir los gastos en las celebración de los eventos y talleres: meriendas, agua, premios... Desde un punto de vista económico, el taller se repetirá tantas veces como se pueda. Ello, obviamente, ajustado al presupuesto que se

asigne al proyecto. A mayor sea el presupuesto más talleres se pondrán realizar.

Una vez que los talleres sean conocidos y valorados por los padres, se podría involucrar a los padres para que entre todos soportemos los gastos derivados de la actividad.

Plan de trabajo

Metodología docente

La programación implica un esfuerzo intelectual y el diseño de un conjunto de pasos que permitan llegar a la solución del problema. La estrategia de resolución se basa en cuatro pasos (Polya, 1965):

1. Entender el problema. Lo peor que puede suceder es que el alumno se embarque en la solución del problema sin entenderlo. Por tanto, la primera fase es entender el problema. Lo que se debe hacer en esta fase es garantizar la comprensión total y absoluta del enunciado.
2. Diseñar una solución. En esta fase es necesario ofrecer una solución al alumno que pueda luego aplicar a problemas similares. También se puede enseñar a resolver un problema similar de menor complejidad, o bien, dividir el problema en problemas más pequeños e ir resolviendo por partes.
3. Ejecutar el plan. El docente debe tener muy poca presencia en esta etapa. Debe limitarse a procurar que el alumno siga el plan propuesto, y evitar que se atasque con temas instrumentales.
4. Revisar. El objetivo de esta fase no sólo es la búsqueda de posibles errores, sino también aprender del proceso. Para ello, se incita al alumno a imaginar otros contextos donde la solución pudiera volver a utilizarse.

La metodología no garantiza alcanzar la solución del problema pero ofrece una forma de organizar la resolución del problema. La forma precisa en el que el problema será resuelto depende de muchos otros factores. Los alumnos deben aprender a trabajar en equipo para resolver el problema, lo que significa aprender a escuchar y a contribuir individualmente lo que cada uno pueda.

Metodología del proyecto

Desde un punto de vista de la ejecución del proyecto, la coordinación del proyecto se realizará con metodologías y herramientas ágiles (Scrum):

1. Reuniones periódicas (daily meetings)
2. Kanban
3. Retrospectivas

A estos efectos, se considerará cada taller como una iteración que nos permita mejorar la forma de enfocar la estimulación y así motivar a los niños de la programación.

Así mismo, esta forma de trabajo nos ayudará a obtener realimentación de los usuarios para pivotar e ir ajustando el servicio en función del interés y las necesidades.

Equipo humano de trabajo y funciones asignadas

Director. José Juan Hernández Cabrera

Coordinador: José Évora Gómez

Docentes. Sergio Falcón, Octavio Roncal, Bycor Sánchez, María Teresa Gálvez, Pedro Marín, Johan Cortes, María del Carmen Sánchez

Asesor educativo. Mario Hernández (Catedrático de Universidad)

Calendario e hitos principales

Inicio del proyecto

- Preparación de la web
- Realización de materiales para los talleres
- Comunicación de la actividad a los colegios

1 marzo. Materiales de los talleres terminados

7 marzo. Acto de presentación del Hacklab y las actividades

14 marzo. Inicio del primer taller

14 mayo. Scratch Day

1 julio. Summer of CodeCraft

19 septiembre. Reanudación de los talleres

19 diciembre. Retrospectiva del hacklab en su primer año

Medios materiales

Para realizar el proyecto, se cuenta con el apoyo de la Escuela de Ingeniería Informática que cedería los huecos ociosos en sus laboratorios para la realización de esta actividad. Estos laboratorios están completamente equipados con ordenadores de escritorio suficientes para la realización del proyecto.

Por otro lado, se usarán con herramientas software como Scratch, Minecraft, Navegadores web..., todos ellos disponibles sin coste para la ejecución del proyecto.

Resultados del proyecto

Resultados esperados

El principal resultado que se espera del proyecto es fundamentalmente iniciar una actividad de formación a niños de 8 a 14 años. Si la actividad tiene éxito, se habrá logrado una experiencia que permitiría repetirla en próximos años.

Con el taller no se alcanzará un dominio de la programación, pero si servirá como una iniciación y como un catalizador en las inquietudes de los niños. En este sentido, se espera introducir pautas en los niños para abordar la resolución de problemas que pueden ser útiles para otro tipo de contenidos como las matemáticas o cualquier actividad científica.

Plan de difusión y diseminación de resultados

Campaña entre los colegios de la isla previa al inicio de los talleres.

Campaña en los medios de comunicación al inicio y durante la ejecución de los talleres. Para realizar esta campaña se contratará a la empresa AE Informaciones.

Redes sociales (Twitter, Facebook) y blog de la cátedra de telefónica.

Video explicativo y cuaderno divulgativo.

Presupuesto económico

Concepto	Importe
Preparación de materiales educativos	1.000 €
Personal. Difusión Realización de la página web Campaña de comunicación Publicación en blog y redes sociales Realización del video explicativo Elaboración del cuaderno	2.000 €
Gastos varios derivados de la organización de los talleres	3.000 €
Total	6.000 €