SISTEMA DE INFORMACION PARA SEGUIMIENTO DE LOS BENEFICIARIOS DE TALENTO DIGITAL

JUAN EDUARDO SALAZAR ARCILA  
Código 506131025

Director:

JORGE ELIECER CAMARGO MENDOZA, PhD

KONRAD LORENZ FUNDACIÓN UNIVERSITARIA

FACULTAD DE MATEMÁTICAS E INGENIERÍAS

PROGRAMA DE INGENIERIA DE SISTEMAS

BOGOTÁ

2015

1. FORMULACIÓN DEL PROBLEMA
   1. Descripción:

En la actualidad en nuestro país tenemos un déficit de 15.000 ingenieros y trabajadores de las TI, lo que muestra la amplia oportunidad laboral en este campo. A partir de esta necesidad surge Talento Digital que es el resultado del trabajo conjunto del Ministerio TIC en alianza con Icetex y la Asociación Colombiana de Ingenieros de Sistemas (ACIS).

[[1]](#footnote-1)Talento Digital es un fondo que busca formar talento humano en TI por medio de convocatorias que ofrecen créditos condonables hasta el 100% de la matrícula para animar a la formación de los colombianos en carreras TI, para impulsar la competitividad, la investigación, la innovación y la proyección internacional del sector TIC de Colombia.

Las TI están constantemente transformando a Colombia y el mundo, ellas apoyan al crecimiento de la calidad de vida de las personas, así mismo, para encontrar un proyecto de vida encaminado a un desarrollo laboral y profesional.

El programa de Talento Digital en la actualidad controla el seguimiento de sus beneficiarios por medio de formatos de Excel lo que hace que sea una labor tediosa al momento de generar reportes y estadísticas con estos datos.

* 1. Área de conocimiento:

El área de conocimiento en la cual se va enfocar el proyecto la ingeniería de software, bases de datos, análisis de sistemas.

* 1. Alcances y delimitaciones:
* Se entregará una aplicación web en la que los funcionarios del programa de Talento Digital puedan realizar seguimiento a los estudiantes.
* La Aplicación podrá ser accedida desde un computador con acceso a internet.
* Solo se permitirá el acceso a usuarios previamente verificados por el programa de Talento Digital.
* Cualquier estudiante beneficiario de Talento Digital podrá crear su perfil.

1. JUSTIFICACIÓN

El proyecto es necesario realizarlo ya que un sistema de seguimiento mejorara esta labor por parte de los funcionarios de Talento Digital que tienen que manejar seguiimiento en la actualidad a mas de 6.000 beneficiarios en toda Colombia. Esta aplicación será una herramienta de gran utilidad para los estudiantes que se encuentran realizando sus estudios en las áreas de tecnologías de la Información (TI) gracias al programa de talento digital impulsado por el Gobierno colombiano y el Ministerio de TIC. Este proyecto será desarrollador utilizando tecnologías web y con los últimos frameworks de desarrollo, para que desde cualquier dispositivo que tenga conexión a Internet pueda beneficiarse con sus múltiples características.

1. OBJETIVOS GENERAL

Desarrollar una aplicación web mediante la cual los funcionarios de Talento Digital puedan llevar el control y seguimiento de todos los beneficiarios en todo el pais.

1. OBJETIVOS ESPECÍFICOS

* Desarrollar una aplicación web que se adapte a cualquier dispositivo.
* Brindar una nueva herramienta que permita mejorar el seguimiento de los beneficiarios Talento Digital.
* Ofrecer a los los funcionarios de Talento Digital un modulo de reportes sobre los beneficiarios de Talento Digital.

1. MARCO REFERENCIAL

* 1. Marco histórico:

Actualmente los beneficiaros de Talento digital llevan un formato en Excel cuando tiene un proyecto que desean realizar dicho formato se envia por medio de correo electronico.

* 1. Marco Teórico

1. **HyperText Preprocessor (PHP)[[2]](#footnote-2)**

HyperText Preprocessor o PHP es un lenguaje interpretado del lado del servidor que se caracteriza por su potencia, versatilidad. Robustez y modularidad. Los programas escritos en PHP son embebidos directamente en el código HTML y ejecutados x el servidor web a través de un intérprete antes de transferir al cliente que lo ha solicitado un resultado en forma de código HTML puro. Al ser un lenguaje que sigue la corriente *open source,* tanto el intérprete como su código fuente son totalmente accesibles de forma gratuita en la red. En concreto, la dirección oficial en la que se puede descargar es [www.php.net](http://www.php.net).

En comparación con otro tipo de tecnologías similares, PHP resulta más rápido, independiente de la plataforma y más sencillo de aprender y utilizar.

Todas estas características han hecho de este lenguaje uno de los que mayor crecimiento ha experimentado en los últimos años, desde su aparición en 1994. Es de destacar especialmente la factibilidad para la conectividad con sistemas gestores de bases de datos a través de un gran número de funciones especializadas. Esa facilidad de conexión ha hecho que PHP sea actualmente uno de los lenguajes más utilizados para la generación de páginas dinámicas, no solo personales si no también portales de empresas y organizaciones. Inicialmente diseñado para realizar poco más que contadores y libros de visita de páginas, en la actualidad php permite realizar una multitud de tareas útiles para el desarrollo web. Por ejemplo, dispone, entre otras, de:

* 1. Funciones de correo electrónico que pueden ser utilizadas para programar completos sistemas de correo electrónico vía web
  2. Funciones de administración y gestión de bases de datos específicas para la mayoría de los gestores comerciales y funciones para conexiones ODBC con bases de datos en sistemas Microsoft.
  3. Funciones de gestión de directorios y ficheros, incluso para la transferencia mediante FTP.
  4. Funciones de tratamiento de imágenes y librerías de funciones graficas
  5. Funciones de generación y lectura de cookies
  6. Funciones para generación de documentos PDF.

PHP, un lenguaje de programación diseñado específicamente para ser usado en la web, es su herramienta para crear páginas web dinámicas. Como es rico en características que facilitan el diseño y programación web. PHP se usa en más de 13 millones de dominios (según la encuesta de netcraft en [www.php.net/usage.php](http://www.php.net/usage.php)). Su popularidad continua creciendo, lo cual significa que debe estar cumpliendo muy bien su función.

Php significa HyperText Preprocessor/Preprocesador de hipertexto. Cuando Rasmus Lendford empezó a desarrollarlo, se llamaba Personal Home Page Tools/ Herramientas para una página de inicio personal. Cuando se desarrolló un lenguaje más completo. El nombre se le cambio para que estuviera más a tono con su funcionalidad expandida.

La sintaxis del lenguaje PHP es parecida a la sintaxis de C, así que si tiene experiencia con C, se sentirá cómodo usando PHP. PHP es incluso más simple que C porque no usa algunos conceptos de la programación de bajo nivel de C porque PHP está diseñado para programar sitios web y no necesita esas capacidades.

La habilidad de PHP para interactuar con bases de datos es particularmente fuerte. PHP puede trabajar con prácticamente todas las bases de datos de las cuales haya escuchado hablar. PHP maneja la conexión con la base de datos y la comunicación con ella. Usted no necesita conocer los detalles técnicos para conectarse a una base de datos o para intercambiar mensajes con ella. Solo debe decir a PHP el nombre de la base de datos y donde esta, y PHP se encarga de los detalles. Se conecta a la base de datos, para sus instrucciones a la base de datos le trae de vuelta la respuesta de la base de datos.

1. **Lenguaje Unificado de Modelado (UML 2.0)**

Cualquier rama de ingeniería o arquitectura ha encontrado útil desde hace mucho tiempo la representación de los diseños de forma gráfica. Desde los inicios de la informática se han estado utilizando distintas formas de representar los diseños de una forma más bien personal o con algún.

Modelo gráfico. La falta de estandarización en la manera de representar gráficamente un modelo impedía que los diseños gráficos realizados se pudieran compartir fácilmente entre distintos diseñadores.

Se necesitaba por tanto un lenguaje no sólo para comunicar las ideas a otros desarrolladores sino también para servir de apoyo en los procesos de análisis de un problema. Con este objetivo se creó el Lenguaje Unificado de Modelado (UML: Unified Modeling Language). UML se ha convertido en ese estándar tan ansiado para representar y modelar la información con la que se trabaja en las fases de análisis y, especialmente, de diseño.

El lenguaje UML tiene una notación gráfica muy expresiva que permite representar en mayor o menor medida todas las fases de un proyecto informático: desde el análisis con los casos de uso, el diseño con los diagramas de clases, objetos, etc., hasta la implementación y configuración con los diagramas de despliegue.[[3]](#footnote-3)

Lenguaje Unificado de Modelado (UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de [modelado](http://es.wikipedia.org/wiki/Modelado) de sistemas de [software](http://es.wikipedia.org/wiki/Software) más conocido y utilizado en la actualidad; está respaldado por el [OMG](http://es.wikipedia.org/wiki/Object_Management_Group) (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y compuestos reciclados.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o [RUP](http://es.wikipedia.org/wiki/RUP)), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, sin embargo, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos. UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.[[4]](#footnote-4)

1. **Sistema de información**

Un sistema de información es un conjunto de componentes que interaccionan entres si para lograr un objetivo en común. Aunque existe una gran variedad de sistemas, la mayoría de ellos pueden representarse a través de un modelo formado por bloques básicos, elementos de entrada, elementos de salida, sección de transformación, mecanismos de control y objetos. Tal y como se ve en la figura, los recursos acceden al sistema a través de los elementos de entrada para ser modificados en la sección de transformación. Este proceso es controlado por el mecanismo de control con el fin de lograr el objetivo marcado. Una vez se la ha llevado a cabo la transformación, el resultado sale del sistema a traves de los elementos de salida

Figura.Modelo general de un sistema

La sociedad actual está llena de ejemplos de sistemas: una máquina expendedora de bebidas, una fábrica de productos manufactureros, un archivador de documentos etc. En el caso de la máquina expendedora, el elemento de entrada corresponderá a la ranura para la introducción de monedas. Una vez están las monedas en el sistema, se comparan con el precio de la bebida seleccionada (objetivo del sistema) mediante el sistema de control. Cuando la cantidad de dinero es introducida en el sistema corresponde con el precio de la bebida. El mecanismo de control cambia la moneda por una bebida, la cual es entregada a través del expendedor de la máquina. De forma similar, es posible representar el resto de los ejemplos mediante los cinco conceptos básicos de sistema.

El concepto de sistema de información ha sido definido de varias maneras por diferentes autores que han investigado sobre el tema; a pesar de todo, podemos decir que todavía no se ha llegado a un consenso generalizado con respecto a su definición.[[5]](#footnote-5)

1. **Función intrínseca del sistema de información**

Un sistema de información es un sistema que reúne, almacena, procesa y distribuye un conjunto de información entre los diferentes elementos que configuran una organización, y entre la organización misma y su entorno.

Las funciones intrínsecas de un sistema de información que están implicadas en la definición que acabamos de enunciar se pueden ilustrar en la siguiente figura.

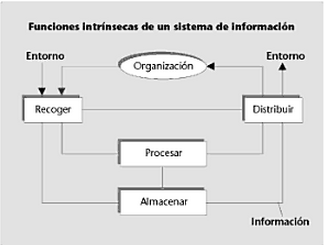


Figura. Función Intrínseca del sistema de información.

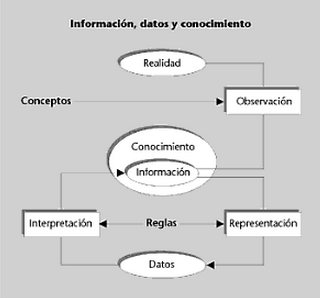
1. **Componentes básicos de un sistema de información**

Los conceptos de sistema de información, y de hecho, de información son conceptos abstractos que se pueden implementar de muchas maneras diferentes en el mundo físico.

Dado que la información se puede considerar como la materia prima de todo el sistema de información, conviene empezar por definirla y, de pasada, por relacionarla con otros conceptos cercanos como el de datos y el de conocimiento.

*La información es conocimiento transferible, recopiladle y procesable que se representa mediante datos almacenados en un soporte.*

La siguiente figura nos puede ser útil para observar la relación que se da entre la información, conocimiento y datos.[[6]](#footnote-6)

**Figura. Información, datos y conocimiento.

1. **Tipos de sistemas de información[[7]](#footnote-7)**

La organización cuenta con sistemas de apoyo a ejecutivos (ESS, por sus siglas en inglés) en el nivel estratégico; sistemas de información gerencial (MIS) y sistemas de apoyo a la toma de decisiones (DSS) en el nivel administrativo; sistemas de trabajo del conocimiento (KWS), sistemas de oficina en el nivel de conocimiento, y sistemas de procesamiento de transacciones (TPS) en el nivel operativo. A su vez, los sistemas de cada nivel se especializan en apoyar a cada una de las principales áreas funcionales.

1. **MVC (Modelo Vista Controlador)**

El patrón MVC es un patrón de arquitectura de software encargado de separar la lógica de negocio de la interfaz del usuario y es el más utilizado en aplicaciones Web, ya que facilita la funcionalidad, mantenibilidad y escalabilidad del sistema, de forma simple y sencilla, a la vez que permite “no mezclar lenguajes de programación en el mismo código”.[[8]](#footnote-8)

MVC divide las aplicaciones en tres niveles de abstracción:

* 1. Modelo: representa la lógica de negocios. Es el encargado de accesar de forma directa a los datos actuando como “intermediario” con la base de datos.
  2. Vista: es la encargada de mostrar la información al usuario de forma gráfica y “humanamente legible”.
  3. Controlador: es el intermediario entre la vista y el modelo. Es quien controla las interacciones del usuario solicitando los datos al modelo y entregándolos a la vista para que ésta, lo presente al usuario, de forma “humanamente legible”.

El funcionamiento básico del patrón MVC, puede resumirse en:

* 1. El usuario realiza una petición.
  2. El controlador captura el evento (puede hacerlo mediante un manejador de eventos – handler -, por ejemplo).
  3. Hace la llamada al modelo/modelos correspondientes (por ejemplo, mediante una llamada de retorno – callback -) efectuando las modificaciones pertinentes sobre el modelo.
  4. El modelo será el encargado de interactuar con la base de datos, ya sea en forma directa, con una capa de abstracción para ello, un Web Service, etc. Y retornará esta información al controlador.
  5. El controlador recibe la información y la envía a la vista.
  6. La vista, procesa esta información pudiendo hacerlo desde el enfoque que veremos en este libro, creando una capa de abstracción para la lógica (quien se encargará de procesar los datos) y otra para el diseño de la interfaz gráfica o GUI.
  7. La lógica de la vista, una vez procesados los datos, los “acomodará” en base al diseño de la GUI - layout – y los entregará al usuario de forma “humanamente legible”.

Programar utilizando MVC consiste en separar la aplicación en tres partes principales. El modelo representa los datos de la aplicación, la vista hace una presentación del modelo de datos, y el controlador maneja y enruta las peticiones [requests] hechas por los usuarios.

  
Figura. Modelo vista controlador.

¿Por qué utilizar MVC? Porque es un patrón de diseño de software probado y se sabe que funciona. Con MVC la aplicación se puede desarrollar rápidamente, de forma modular y sostenible. Separar las funciones de la aplicación en modelos, vistas y controladores hace que la aplicación sea muy ligera. Estas características nuevas se añaden fácilmente y las antiguas toman automáticamente una forma nueva.

El diseño modular permite a los diseñadores y a los desarrolladores trabajar conjuntamente, así como realizar rápidamente el prototipado. Esta separación también permite hacer cambios en una parte de la aplicación sin que las demás se vean afectadas.[[9]](#footnote-9)

1. DISEÑO METODOLÓGICO PRELIMINAR

El problema se solucionará por medio de pequeños entregables que se validarán constantemente con Talento Digital, basándose en algunos elementos de las metodologías **Extreme Programming y Scrum.**

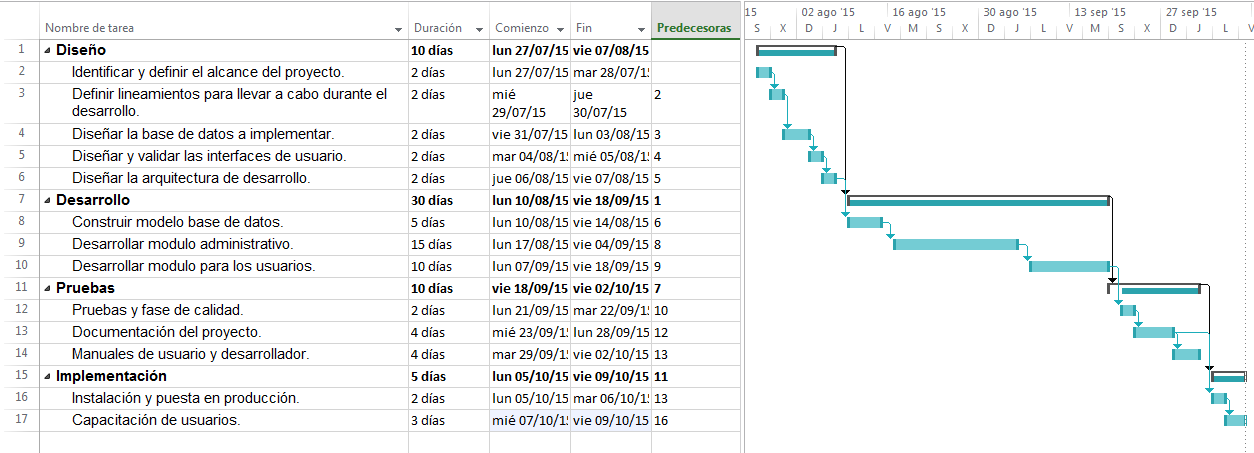
|  |  |
| --- | --- |
| Actividad | Entregable |
| Especificación de requerimientos | Documento Especificación de requerimientos. |
| Análisis y diseño | Diseños de pre Pantallas (Mockups). Diseño de base de datos.  Diseño de Api Rest. |
| Construcción | Software web.  Código Fuente.  Manual del programador.  Manual de usuario. |
| Pruebas | Informe pruebas unitarias.  Informe pruebas integración.  Informe pruebas sistema.  Informe pruebas aceptación. |

1. PROPUESTA DE LA SOLUCIÓN
   1. Descripción

El sistema de información web permitirá a los funcionarios consultar la informacion de los beneficiarios de Talento Digital, realizar filtros de la informacion para hacerla mas facil de manejar, generar reportes de la informacion, los beneficiarios de Talento Digital podran crear su cuenta y actualizar sus datos.

* 1. Listado de entregables
* Sistema de información en la Web.
* Código Fuente.
* Manual de usuario y programador
* Base de datos.

1. CRONOGRAMA



1. BIBLIOGRAFÍA
   1. Conde, jesus, video tutorial php-Mysql Dvd español,2005, España: intercambios virtuales, 1.03 GB.
   2. Gilfillan,Ian,MySQL, España: Anaya multimedia, 841 paginas.
   3. Aguilar , luis joyanes, programación orientada a objetos, 2da edición, España: magraw-hill, 163 paginas
   4. Alarcon, raul, diseño orientado a objetos con uml, España: grupo eidos,117 paginas
   5. Batiz, juan de dios, desarrollo orientado a objetos con UML, universidad de castilla- La mancha, 38 paginas
   6. Kendall, Kenneth y kendall, julie, análisis y diseño de sistemas d einformacion, 3ra edición, USA: prentice-hall,923 paginas
   7. León, Gonzalo serrano, ingeniería de sistemas de software, 1ªedicion, España: isdefe, 211 paginas
   8. Pressman, roger, ingeniería de software: un enfoque práctico, 6ª edición, mexico: McGraw-Hill, 900 paginas
2. INFOGRAFIA
3. **Angular JS**  
   Documentación Publica oficial  
   <https://drive.google.com/drive/folders/0BxgtL8yFJbacQmpCc1NMV3d5dnM>  
   Fecha de consulta: 1 Julio de 2015

1. Definición tomada desde <http://talentodigital.mintic.gov.co/625/w3-channel.html> 10 de mayo de 2015 [↑](#footnote-ref-1)
2. Valade, janet. PHP y MySQL para dummies, pg. 15-17 [↑](#footnote-ref-2)
3. Hernández Orallo, Enrique. “El lenguaje de modelado de unificado”, pg. 1 [↑](#footnote-ref-3)
4. es.wikipedia.org/wiki/Lenguaje\_Unificado\_de\_Modelado [↑](#footnote-ref-4)
5. Fernández Alarcón, Vicenç. Desarrollo de sistemas de información: Una metodología basada en el modelado, pg. 11 [↑](#footnote-ref-5)
6. Pastor i Collado, Joan Antoni. Concepto de sistema de información en una organización,pg.7-9 [↑](#footnote-ref-6)
7. http://biblioteca.itson.mx/oa/dip\_ago/introduccion\_sistemas/p7.htm [↑](#footnote-ref-7)
8. http://www.eugeniabahit.com/mvc/ [↑](#footnote-ref-8)
9. http://book.cakephp.org/1.3/es/view/890/Entendiendo-Modelo-Vista-Controlador [↑](#footnote-ref-9)