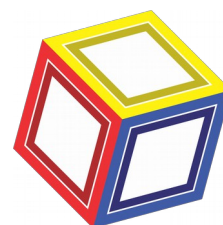
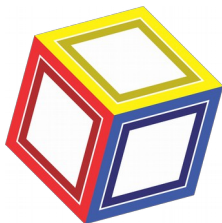


Plan de Gestión de Proyecto

Proyecto: Bestnid
Revisión 05



Oct-2014

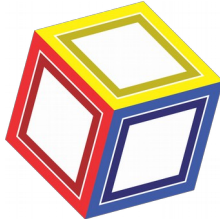


Ficha del documento

Fecha	Revisión	Autor	Verificado
10/10/2014	01	Cube's Crew	
12/10/2014	02	Cube's Crew	
13/10/2014	03	Cube's Crew	
28/10/2014	04	Cube's Crew	
24/11/2014	05	Cube's Crew	

Documento validado por las partes en fecha: [Fecha]

Por el cliente	Por la empresa suministradora
Aclaración	Aclaración



1 Introducción

1.1 Resumen del Proyecto

1.1.1 Propósito, alcance y objetivos.

El fin del presente es formalizar la planificación del proyecto que tendrá como resultado final un software capaz de atender un sistema de subastas online.

El proyecto abarca todas las etapas de desarrollo del software incluyendo desde la elicitación de requerimientos hasta las pruebas de aceptación del producto y sus tareas de mantenimiento una vez finalizado el desarrollo del software.

1.1.2 Supuestos y restricciones

El proyecto estará finalizado el 12 de Diciembre del año actual.

1.1.3 Entregables del proyecto

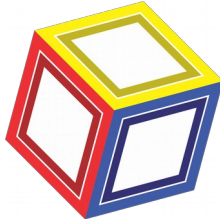
El proyecto tiene como única entrega tangible toda la documentación que corresponde a los estándares de documentación. Se presentarán a los clientes prototipos del producto en los días 14 de Noviembre y 28 de Noviembre del corriente año. La entrega final del producto completo se realizará el 12 de Diciembre.

1.1.4 Calendario y resumen del presupuesto

El tiempo de desarrollo total del proyecto está planeado por 6 meses. El presupuesto total será de \$19620, que consiste en las horas de trabajo de todos los integrantes del equipo.

2 Documentos referenciados

Referencia	Titulo	Fecha	Autor
	IEEE Std 1058-1998	1998	IEEE



3 Definiciones y acrónimos

Se refiere al producto a desarrollar como la **aplicación**.

Historia de usuario: requerimiento funcional del software expresado en forma coloquial.

Product Backlog o Pila de Producto: colección de todas las historias de usuarios.

Framework: estructura que sirve de soporte o base para la organización y desarrollo de [software](#).

Webhost: servicio que provee a los usuarios de Internet un sistema para poder almacenar información accesible vía web.

Git: es un software de control de versiones pensado para aplicaciones web.

Wiki: Es una aplicación web que le permite a los usuarios añadir, modificar o borrar contenidos en colaboración con otros.

Demo: Es una versión de funcionalidad reducida del producto.

4 Organización del proyecto

4.1 Interfaces externas

La interacción con los dueños la realiza todo el equipo mediante entrevistas solicitadas por el miembro del equipo de desarrollo designado como Scrum Master.

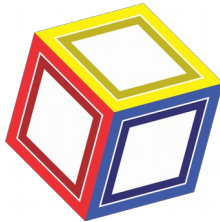
4.2 Estructura interna

Se utilizará la metodología ágil Scrum para el desarrollo de la aplicación. Los miembros del proyecto aportarán en todas las actividades como dicta la filosofía de dicha metodología. Los diferentes miembros del equipo de desarrollo tomarán alguno de los siguientes roles: Propietario, Scrum Master, miembro del Scrum Team. Además, la estructura interna será Descentralizada democrática.

4.3 Roles y responsabilidades

Los diferentes roles que hay en el proyecto son:

- **Propietario:** Su responsabilidad es establecer la dirección del proyecto y definir prioridades.
- **Scrum Master:** Su trabajo consiste en lograr que el proyecto progrese de acuerdo a las prácticas, valores y reglas de SCRUM.
- **Miembro del Scrum Team:** Responsable de transformar la Pila de Producto de la iteración en un incremento de la funcionalidad del software



5 Planes de administración del proceso

5.1 Plan inicial

5.1.1 Plan del personal

El personal disponible para el proyecto son los cuatro integrantes del equipo Cube's Crew. Uno cumple la función de Scrum Master y tres conforman el Scrum Team. Todos los miembros estarán presentes durante todo el proyecto.

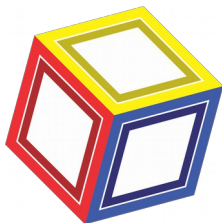
5.1.2 Plan de adquisición de recursos

Se utilizarán una variedad de herramientas en su versión gratuita, las cuales son totalmente suficientes para los fines del proyecto.

- **Ruby on Rails.**
- **Github:** Webhost para alojar repositorios de proyectos y que automatiza el versionado de los mismos utilizando Git.
- **Heroku:** Webhost para ejecutar la aplicación y poder ser accedida desde cualquier navegador.
- **Pivotal Tracker:** Herramienta para gestionar las historias de usuario dentro del marco de planificación de desarrollo.
- **PostgreSQL:** Base de datos.

5.1.3 Plan de entrenamiento del personal del Proyecto

Los miembros cuentan con una formación inicial en Ruby on Rails. Complementarán su formación y aprenderán a utilizar todas las otras herramientas mencionadas previamente en 5.1.2 en el curso del desarrollo del proyecto.



5.2 Plan de trabajo

5.2.1 Principales actividades del proyecto

Obtención y análisis de requerimientos:

Realizar entrevistas para averiguar las funcionalidades pedidas por el cliente.

Planificación:

Elaboración de las historias de usuario y calificación mediante peso según el tiempo estimado de desarrollo.

Diseño:

Desarrollo de los prototipos de la solución.

Codificación:

Programación de la solución (funcionalidades).

Pruebas de Verificación:

Comprobación del funcionamiento del sistema.

Pruebas de Validación:

Comprobación del cumplimiento de los requerimientos.

Capacitación

5.2.2 Asignación de esfuerzo

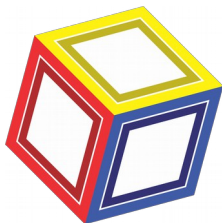
Actividad	Cantidad	Esfuerzo (hs) Unitario	Esfuerzo (hs) Subtotal
Obtención de requerimientos	4	2	8
Análisis de requerimientos	4	2	8
Planificación	4	1	4
Diseño	4	14	56
Implementación	4	17	68
Pruebas de Verificación	4	17	68
Pruebas de Validación	2	3	6
Capacitación	1	4	4

5.2.3 Asignación de presupuesto

	Cantidad	Precio Unitario	Precio
Horas	222	90	19980
Total			19980

El presupuesto total del proyecto será:

Pesos diecinueve mil novecientos ochenta (\$19980,00)



5.3 Plan de control

5.3.1 Plan de control de requerimientos

No aplica.

5.3.2 Plan de control de calendario

No aplica

5.3.3 Plan de control de presupuesto

No aplica.

5.3.4 Plan de control de calidad

No aplica.

5.3.5 Plan de informe

No aplica.

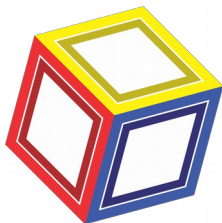
5.3.6 Plan de recolección de métricas

No aplica.

5.4 Plan de administración de riesgo

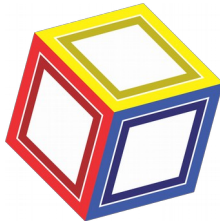
Id Riesgo	Nombre	Probabilidad	Impacto	Responsable	Estado
005	Borrado de la base de datos	BAJA	ALTO	Mantenimiento	
007	Planificación demasiado optimista	MEDIA	MEDIO	Equipo	
017	Cambio de Requisitos	MEDIA	MEDIO	Cliente	

Id Riesgo	Nombre	Probabilidad	Impacto	Responsable	Estado
002	Indisposición de un miembro del equipo	BAJA	MEDIO	Equipo	
003	Caída del servidor	BAJA	MEDIO	Servidor	
011	Pérdida de recursos	BAJA	MEDIO	Equipo	
013	Imposibilidad de Implementación	MUY BAJA	ALTO	Equipo	



Id Riesgo: 005	Nombre : Borrado de la base de datos	
	Fecha:	
	Descripción : La base de datos con la información de los usuarios registrados es borrada.	
Probabilidad: BAJA		
Impacto: ALTO		
Responsable: Mantenimiento		Clase: Producto
Estrategia de Mitigación (Anulación/Minimización): Restringir el acceso a terceros y elegir un servidor seguro.		
Plan de Contingencia : Hacer copias de seguridad periodicamente.		

Id Riesgo: 007	Nombre : Planificación demasiado optimista	
	Fecha:	
	Descripción : Se intentan abarcar más historias de usuario en un ciclo de las que se pueden solucionar.	
Probabilidad: MEDIA		
Impacto: MEDIO		
Responsable: EQUIPO		Clase: Proyecto
Estrategia de Mitigación (Anulación/Minimización): Darle el peso correcto a cada Historia de Usuario.		
Plan de Contingencia : La demo se verá retrasada. Se dejarán Historias de Usuario para implementar en la próxima Demo.		



Id Riesgo: 017	Nombre : Cambio de requisitos	
	Fecha:	
	Descripción : Aparición de nuevos requisitos o cambio de las funcionalidades pedidas por el cliente.	
Probabilidad: MEDIA		
Impacto: MEDIO		
Responsable: EQUIPO		Clase: Producto y proyecto
Estrategia de Mitigación (Anulación/Minimización): Los prototipos servirán para capturar a tiempo dichos cambios para reducir el impacto.		
Plan de Contingencia : Habrá mayor trabajo para el próximo ciclo, lo que podría retrasar el proyecto.		

5.5 Plan de liberación de proyecto

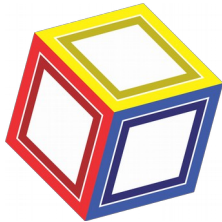
No aplica.

6 Planes de procesos técnicos

6.1 Modelo de proceso

El modelo empleado será el establecido por la metodología Scrum. Consiste en realizar ciclos completos de desarrollo con una duración relativamente corta. Durante cada ciclo se realizarán las distintas tareas en el siguiente orden:

- **Definición de requerimientos:** En cada ciclo habrá un relevamiento para definir los requerimientos, éste consiste en un análisis de requerimientos en el primer ciclo de desarrollo, y en un estudio de la información obtenida en la anterior prueba de validación para los restantes ciclos.
- **Diseño:** Se realiza un prototipo de la implementación de cada funcionalidad escogida en la etapa anterior.
- **Codificación:** Se implementan los prototipos diseñados realizando primero las pruebas de verificación para las funcionalidades y luego desarrollandolas.
- **Prueba de validación:** Se prepara el lanzamiento de una nueva Demo para que sea comprobada por el cliente.



6.2 Métodos, herramientas y técnicas

Se utilizará **Ruby on Rails**, un framework de aplicaciones web, para implementar la aplicación. Se trabajará con la metodología Test Driven Development o Desarrollo Orientado a Pruebas utilizando la herramienta RSpec.

6.3 Plan de infraestructura

Cada miembro del personal cuenta con su computadora personal que utilizará para realizar los aportes en las distintas actividades del proyecto. Como se trabajará utilizando muchas herramientas web (GitHub, Pivotal Tracker) es necesaria la conexión a internet para realizar la sincronización entre todos los aportes. La conexión a internet es imprescindible a la hora de las pruebas de validación con los clientes (a través de Heroku).

6.4 Plan de aceptación del producto

Las pruebas de validación del software se realizarán demostrando el uso de la aplicación en una Demo al cliente, quién analizará los aspectos positivos y negativos del proyecto. Estos datos serán utilizados para mejorar algunos aspectos en el nuevo ciclo de desarrollo.

7 Plan de procesos de apoyo

7.1 Plan de administración de configuración

No aplica.

7.2 Plan de pruebas

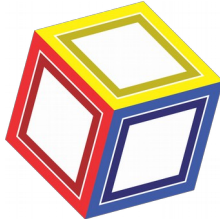
No aplica.

7.3 Plan de documentación

El proyecto cuenta con una amplia documentación conformada por: El presente documento, la Especificación de Requerimientos del Software del proyecto, las entrevistas y cuestionarios llevados a cabo, la documentación de la interfaz, y la documentación del código junto con la wiki del proyecto halladas en Github.

7.4 Plan de aseguramiento de calidad

No aplica.



7.5 Plan de revisiones y auditorías

No aplica.

7.6 Plan de resolución de problemas

No aplica.

7.7 Plan de administración de terceros

No aplica.

7.8 Plan de mejoras en el proceso

No aplica.

8 Planes adicionales

Plan de mantenimiento del proyecto

- Limpieza y compresión de la base de datos automática una vez al mes.