

HABILITACIÓN PROFESIONAL

AGRADECIMIENTOS

Me gustaría expresar mi más profundo agradecimiento a todas aquellas personas que me han facilitado la posibilidad de llegar hasta esta instancia de mi carrera y hacer posible la realización de este proyecto. Han sido muchas estas personas, pero me gustaría destacar a algunas.

En primer lugar quiero nombrar a mi familia, que siempre han apoyado mi labor como estudiante y me han empujado en los momentos más duros.

Al Ing. Sergio Rodríguez por su predisposición, tiempos, dedicación y enseñanzas brindadas haciendo posible que mi proyecto tome forma y contenido, además del perfeccionamiento de su estructura.

Al personal administrativo del CIIDEPT (Centro de innovación e información para el Desarrollo Educativo, Productivo y Tecnológico) por el tiempo y la constante información brindada para la correcta elaboración del sistema. A los directivos por solicitarme colaborar en la elaboración de este sistema de reservas, donde tuve la oportunidad de aprender mucho acerca de las nuevas tecnologías de desarrollo y poner en práctica lo estudiado en la universidad. A Luis González del equipo de sistemas, quien me acompañó y guió durante el desarrollo compartiendo sus conocimientos técnicos y experiencia como desarrollador.

TABLA DE CONTENIDO

1 GENERALIDADES Y PRESENTACION	8
1.1 INTRODUCCIÓN.....	8
1.2 OBJETIVOS.....	9
1.2.1 OBJETIVO GENERAL	9
1.2.2 OBJETIVOS ESPECÍFICOS.....	9
1.3 ALCANCE DEL PROYECTO	10
1.4 METODOLOGÍA USADA	11
1.4.1 MARCO TEÓRICO DE RUP	11
1.4.2 ESTRUCTURA DEL PROYECTO	12
1.5 ESTADO DEL ARTE	13
1.6 CARACTERÍSTICAS DE LAS TECNOLOGÍAS WEB.....	15
2 FASE INICIAL	17
2.1 ACTA DE CONSTITUCIÓN DEL PROYECTO.....	17
2.2 GESTIÓN DE LOS INTERESADOS.....	17
2.3 GESTIÓN DE CRONOGRAMA	17
2.3.1 DEFINIR ACTIVIDADES.....	17
2.3.2 DEFINIR LA CALENDARIZACIÓN	18
2.3.3 DIAGRAMA DE GANTT DEL PROYECTO.....	19
2.4 GESTIÓN DE RIESGOS	20
2.4.1 ANÁLISIS DE RIESGOS.....	20
3 FASE DE ELABORACIÓN	22
3.1 ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE.....	22
3.2 MODELO DE CASOS DE USO	22
3.2.1 MODELO DE DOMINIO.....	23
3.2.2 DIAGRAMA DE CASOS DE USO.....	24
3.2.3.DOCUMENTACIÓN DE LOS CASOS DE USO	25
3.2.4 DIAGRAMAS DE SECUENCIA	36
3.3 ARQUITECTURA DE SOFTWARE.....	42
3.3.1 MARCO TEÓRICO – ARQUITECTURA DEL SOFTWARE	42
3.3.2 REPRESENTACIÓN DE VISTAS & DIAGRAMAS UML.....	44
3.4 ARQUITECTURA DE HARDWARE.....	47
3.5 MODELO DE DATOS.....	48
4 FASE DE PRODUCCIÓN.....	50
4.1.LINGUAJES DE PROGRAMACIÓN UTILIZADOS	50
4.1.1 PHP	50
4.1.2.CSS.....	52
4.1.3.MySQL.....	53
4.1.4.APACHE.....	54
5 FASE DE TRANSICIÓN.....	56
5.1 PLAN DE PRUEBAS.....	56
5.1.1 MARCO TEÓRICO DE LAS PRUEBAS DE SISTEMA	56

5.1.2 plan de PRUEBAS DE SISTEMA.....	56
5.1.3 ESQUEMA DE Las PRUEBAS	58
5.1.4 PASOS DEL PLAN DE PRUEBAS.....	58
5.2 REPORTES.....	60
5.2.1 INTRODUCCIÓN	60
5.2.2 reportes de ventas.....	
60	
6 CONCLUSIONES	65
7 ANEXOS.....	67
7.1 ANEXO A – ACTA DE CONSTITUCIÓN DEL PROYECTO	67
7.1.1 INFORMACIÓN GENERAL	67
7.1.2 ANTECEDENTES.....	67
7.1.3 JUSTIFICACIÓN DEL PROYECTO.....	68
7.1.4 IDENTIFICACIÓN DE LOS INTERESADOS	68
7.1.5 REQUISITOS QUE SATISFACEN LAS NECESIDADES, DESEOS Y EXPECTATIVAS DE LOS INTERESADOS.....	69
7.1.6 ALINEAMIENTO DEL PROYECTO A LOS OBJETIVOS ESTRATÉGICOS	70
7.1.7 GERENTE DEL PROYECTO Y SU NIVEL DE AUTORIDAD.....	70
7.1.8 RESTRICCIONES DE LA ORGANIZACIÓN	71
7.1.9 SUPUESTOS (ASUNCIONES) DE LA ORGANIZACIÓN.....	71
7.1.10 PRESUPUESTO	72
7.1.11 PROGRAMA DE HITOS Y ENTREGABLES DEL PROYECTO	73
7.1.12 INTEGRANTES DEL EQUIPO DEL PROYECTO, ROLES.	73
7.1.13 FIRMA DE AUTORIZACIÓN DEL ACTA DE CONSTITUCIÓN.....	73
7.2 ANEXO B – ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE	74
7.2.1 INTRODUCCIÓN	74
7.2.2 PROPÓSITO	74
7.2.3 ALCANCE	74
7.2.4 OBJETIVOS	75
7.3 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS.....	76
7.3.1 DEFINICIONES	76
7.3.2 REFERENCIAS	77
7.3.3 VISIÓN GENERAL DE LA ERS.....	77
7.3.4 DESCRIPCIÓN GENERAL	78
7.3.5 REQUISITOS DE INTERFACES EXTERNAS	80
7.3.6 INTERFACES DE HARDWARE	80
7.3.7 INTERFACES DE SOFTWARE	81
7.3.8 REQUISITOS FUNCIONALES.....	81
7.3.9 REQUISITOS NO FUNCIONALES.....	84
7.3.10 LIMITACIONES DE HARDWARE.....	86
7.3.11 LIMITACIONES DE SOFTWARE	86
7.3.12 OTRAS RESTRICCIONES.....	87
7.3.13 ATRIBUTOS DEL SISTEMA	87
7.4 ANEXO C – PLANTILLA DE PRUEBA DE SISTEMA.....	89

GENERALIDADES Y PRESENTACIÓN

1 GENERALIDADES Y PRESENTACION

1.1 INTRODUCCIÓN

Este trabajo presenta un enfoque práctico para la elección y adecuación de metodologías ágiles de desarrollo de software a un proyecto real: El desarrollo de un sistema de agenda de reservas para el Centro de innovación e información para el Desarrollo Educativo, Productivo y Tecnológico (CIIDEPT). Hoy en día, el establecimiento, cuenta con espacios físicos disponibles para diferentes actividades orientadas a la educación, como por ejemplo, capacitaciones docentes, charlas, talleres y/o clases teóricas y prácticas, las cuales tienen como objetivo potenciar y desarrollar los conocimientos teóricos adquiridos y fundamentalmente fortalecer capacidades en el manejo de los instrumentos y materiales con los que se trabajan en estos espacios.

Este proyecto nace ante la necesidad de gestionar los espacios físicos disponibles para brindar una mejor asignación de los mismos. En un principio, los eventos se registraban manualmente en una planilla de excel, el proceso manual y la gran cantidad de eventos diarios generaba muchos errores y por lo tanto conflictos en la correcta asignación de los recursos. Además la búsqueda y modificación de los eventos era un trabajo agotador debido a la gran cantidad de registros y a la carencia de filtros de búsqueda correctos. Es por esto que se decidió crear una plataforma web que se utilice para administrar de manera efectiva la asignación de las aulas y recursos que dispone el establecimiento. El mismo brindará toda la información necesaria sobre la disponibilidad y capacidades de las aulas para su reserva y asignación de recursos.

1.2 OBJETIVOS

La formulación de este proyecto atiende a la necesidad de contrarrestar y/o minimizar el impacto negativo generado en la institución, al momento de realizar la asignación de un aula, ofreciendo la posibilidad de administrar de una manera más eficiente los espacios disponibles. Ante esta situación, nace la necesidad de diseñar e implementar una plataforma web que permita dar una solución a esta problemática.

1.2.1 OBJETIVO GENERAL

Crear una plataforma web que se utilice para asignar los espacios físicos disponibles de acuerdo a las solicitudes de los diferentes organismos y establecimientos que requieren el uso de un espacio y los recursos necesarios para realizar sus respectivas actividades educativas. Además que brinde toda la necesaria sobre disponibilidad, cupos y recursos disponibles para cada aula.

1.2.2 OBJETIVOS ESPECÍFICOS

- Los usuarios podrán asignar los espacios disponibles de manera rápida y sencilla.
- Se podrán visualizar los diferentes eventos para cada espacio físico, a través de un calendario intuitivo, el cual muestra el detalle de todos los eventos registrados.
- Mantener el orden de los recursos y los espacios físicos del establecimiento.
- Evitar la superposición de eventos reservados para la misma fecha y hora, solo asignando los espacios y recursos que se encuentren disponibles.
- Evitar conflictos generados por asignar espacios o recursos no disponibles en una misma fecha.
- Los clientes no tendrán que perder mucho tiempo en buscar la información acerca de los cursos o eventos que se realizan en el día, ya que el sistema muestra el listado de los eventos del día a través de pantallas que se encuentran ubicadas en el patio central del establecimiento.

- Lograr una búsqueda rápida y efectiva de los eventos para visualizar los detalles correspondientes y/o la modificación de dichos eventos.

1.3 ALCANCE DEL PROYECTO

El sistema permitirá gestionar toda la información relacionada a la reserva de espacios físicos del CIIDEPT, así como la asignación de recursos y el registro de los datos de la persona o institución que solicita la reserva.

También se podrá el tipo de evento, para la publicación de los eventos públicos, y de esta manera lograr que los docentes y personas en general puedan informarse de manera rápida e independiente de los eventos del día a través de las pantallas de información instaladas en el establecimiento.

Fecha esperada de entrega del proyecto: 2 de Diciembre de 2016

1.3.1 CARACTERÍSTICAS DE LAS TECNOLOGÍAS WEB

Para avanzar en este proyecto es necesario hacer un repaso de los conceptos sobre los que se apoya el mismo.

Hoy en día con el avance de las tecnologías orientadas a la web nos permite realizar millones de operaciones diarias. Motivo por el cual cabe destacar algunas de las ventajas que nos brinda internet, para realizar las diferentes gestiones de reservas a través de la nube.

Disponibilidad. Los datos están disponibles a cualquier hora del día, sin dependencia del horario de apertura de la institución. En el caso del Sr. ministro de educación esto es una gran ventaja ya que puede estar informado de todas las actividades que se realizan, tanto internas como públicas.

Integridad. la información brindada por el sistema está completa y con todos los detalles referidos a los eventos.

Rapidez. El acceso a la información se realiza sin demora o tramitaciones

especiales, por lo que puede obtener lo que necesita al instante.

Mejor atención al público. el sistema nos permite realizar consultas de manera rápida y efectiva sobre las necesidades o consultas de los clientes, como así también gracias a Internet se puede mostrar información a través de dispositivos inteligentes, ubicados en lugares estratégicos, con motivo de descentralizar las consultas en las áreas de administración.

1.4 METODOLOGÍA USADA (SCRUM)

Como metodología de desarrollo se optó por una metodología ágil, en este caso fue Scrum. Iremos definiendo los distintos artefactos que intervienen en esta Metodología como así también la forma en que fue adaptado para nuestro Team.

El término “Scrum” (melé), relacionando el desarrollo exitoso de productos con el juego del rugby en el que un equipo auto-organizado (auto gestionado) se mueve junto por el campo de desarrollo de productos.

Hoy en día Scrum es usado por empresas de todos los tamaños tales como Yahoo!, Microsoft, Google, Lockheed Martin, Motorola, SAP, Cisco, entre otros. Ellos dicen haber obtenido mejoras sustanciales, y en algunos casos una completa transformación de la productividad y la moral.

El desarrollo se estructura en ciclos de trabajo llamados Sprint, los cuales no deben durar más de cuatro semanas cada una (siendo dos semanas la duración más habitual), están acotados en el tiempo, normalmente los equipos Scrum escogen una duración de Sprint y la mantienen para todos sus Sprint hasta que mejoran y pueden emplear ciclos más cortos. Al principio de cada Sprint, se selecciona los elementos (peticiones del cliente) de una lista priorizada. El equipo acuerda un objetivo colectivo respecto a lo que creen que podrán entregar al final del Sprint, algo que sea tangible y que estará “terminado” por completo.

Durante el Sprint no se podrán añadir nuevos elementos; Scrum se adapta a los cambios en el siguiente Sprint.

Todos los días el Equipo se reúne brevemente para inspeccionar su progreso y ajustar los siguientes pasos necesarios para completar el trabajo pendiente. Al final del Sprint, el

Equipo revisa el Sprint con los diferentes Stakeholders (interesados e involucrados en el producto) y realiza una demostración de lo que han desarrollado. Se obtiene feedback que podrá ser incorporado en el siguiente Sprint. Scrum enfatiza un producto “funcionando” al final del Sprint que esté realmente “terminado”. En el caso del software, esto significa un sistema que está integrado, testado, con la documentación de usuario generada y potencialmente entregable.

1.4.1 PILAS DEL PRODUCTO

La pila de producto es el corazón de Scrum. Es, básicamente, una lista priorizada de requisitos, o cosas que el cliente quiere, llamamos a esto elementos de la Pila. Pueden incluir los siguientes campos:

- ID – un identificador único.
- Nombre – una descripción corta de la historia. Por ejemplo, “Ver tu historial de transacciones”.
- Importancia o prioridad– el ratio de importancia que el Dueño de Producto da a esta historia.
- Estimación inicial – la valoración inicial del Equipo acerca de cuánto trabajo es necesario para implementar el requerimiento, comparado con otros. La unidad usualmente corresponde a “días-persona ideales”.

1.4.2 PLANIFICACIÓN DEL SPRINT

El propósito de la planificación de Sprint es proporcionar al equipo suficiente información como para que puedan trabajar en paz y sin interrupciones durante unas pocas semanas, y de esta manera ofrecer al cliente un Producto suficiente confianza, produce, concretamente:

- Una meta de Sprint.
- Una lista de miembros (y su nivel de dedicación, si no es del 100%).
- Una Pila de Sprint (lista de requerimientos incluidos en el Sprint).
- Una fecha concreta para la Demo del Sprint.
- Un lugar y momento definidos para el Scrum Diario.

Los usuarios deberían participar de la planificación del sprint, principalmente porque cada historia contiene tres variables que son muy dependientes unas de otras, el alcance, la estimación y la importancia. El alcance y la importancia los fija el cliente que solicita el Producto. La estimación la proporciona el equipo de desarrollo.

Durante una planificación de Sprint, las tres variables antes mencionadas, sufren un ajuste fino y continuo a través del diálogo cara a cara entre el equipo y el usuario.

1.4.3 SCRUM DIARIO (Daily Scrum)

El Scrum diario es esencialmente el lanzamiento, donde todo el mundo decide por dónde va a empezar a trabajar.

Empiezan exactamente a su hora, cada día en el mismo sitio. No hay nada mejor que hacerlo frente al tablón. Normalmente se hacen las reuniones de pie, ya que eso reduce el riesgo de sobrepasar los 15 minutos. Normalmente se actualiza el tablón de tareas durante los Scrum diarios. Conforme cada persona describe lo que hizo el día anterior y lo que hará hoy, mueve los post-it en el tablón. Conforme describe elementos no planificados, pone un pos-it nuevo para cada uno de ellos. Conforme actualiza sus estimaciones, escribe una nueva estimación en el post-it correspondiente y tacha la anterior estimación. A veces el Scrum Master hace todo esto mientras los demás hablan.

Desventaja de Scrum por las tardes: cuando llegas al trabajo por la mañana, tienes que acordarte de qué le dijiste a la gente sobre lo que deberían hacer hoy.

Desventaja de los Scrum por las mañanas: cuando llegas al trabajo por la mañana, debes acordarte de qué hiciste ayer para informar sobre ello hoy.

1.4.4 DEMO DE SPRINT

Una demo de Sprint bien ejecutada, aunque parezca poco espectacular, tiene un efecto muy profundo:

- El equipo obtiene reconocimiento por sus logros. Se sienten bien.
- Otras personas se enteran de lo que está haciendo el equipo.
- La demo consigue feedback vital de los interesados.

- Las demos son (o deberían ser) un evento social donde diferentes equipos pueden interactuar unos con otros y discutir su trabajo. Esto es muy valioso.
- Hacer una demo fuerza al equipo a acabar realmente las cosas y entregarlas (incluso aunque sea sólo en entorno de pruebas).

Consejos:

1. Asegurarse de presentar claramente el objetivo del Sprint. Si hay personas en la demo que no saben nada sobre el producto, tomarse un par de minutos para describirlo.
2. No perder mucho tiempo preparando la demo, concentrarse en mostrar código funcionando.
3. Mantener la demo a nivel de negocio, dejar los detalles técnicos aparte. Concentrarse en “qué hemos hecho” en lugar de “cómo lo hemos hecho”.
4. En la medida de lo posible, dejar que la audiencia pruebe el producto por sí misma.
5. No mostrar un montón de pequeños errores solucionados y funcionalidades triviales. Mencionarlas, pero no mostrarlas, ya que normalmente se tarda mucho y desvía la atención de los requerimientos más importantes.

1.4.5 RETROSPECTIVAS DEL SPRINT

Lo más importante de las retrospectivas es asegurarse de que tienen lugar. Las retrospectivas son extremadamente útiles. De hecho, es el segundo evento más importante de Scrum (siendo el primero la reunión de planificación de Sprint) ya que “Es la mejor oportunidad para mejorar, las buenas ideas surgen durante la retrospectiva y a todo el mundo se le permite contribuir y discutir las ideas”.

Sin las retrospectivas, el equipo seguirá cometiendo los mismos errores una y otra vez.

Consejos:

El formato general varía un poco, pero sería algo como así:

- Se reserva 1-3 horas, dependiendo de cuánta discusión esperemos encontrar. Será conveniente tener una discusión sin interrupciones.
- Participantes: el usuario o cliente, el equipo y SCRUM Master.

- El Scrum Master muestra la Pila de Sprint y, con ayuda del equipo, resume el Sprint. Eventos importantes, decisiones, etc.
- Se hace “la ronda”. Cada persona tiene una oportunidad de decir, sin ser interrumpida, qué piensan que ha ido bien, que podría haber ido mejor y que piensan que debería hacerse de forma diferente en el próximo Sprint.
- Se observa la velocidad estimada frente a la real. Si hay una gran diferencia, se intenta analizar por qué.
- Cuando el tiempo casi se ha acabado, el Scrum Master trata de resumir las sugerencias concretas sobre qué puede hacerse mejor el próximo Sprint.

2 CÓMO LLEVAR A CABO

Para poder conocer de dónde provenía la mínima información que requería este sistema y los pasos que harían falta para llevarlo a cabo se decidió atravesar por todas las fases de desarrollo de un proyecto.

Iniciación: Consistía en el análisis de Requerimientos, recopilación de información, Identificación de los interesados, investigación de los procesos de negocio existentes, etc.

Planificación: Había que pensar cuánto tiempo por día había que dedicarle al proyecto, cuanta gente estaría afectada a este desarrollo, definir los hitos que hacían falta para mostrar los avances en etapas tempranas, Asignar roles al equipo, etc.

Ejecución: Luego de haber decidido todo lo anterior se comenzaría a llevar a cabo el sistema. En esta etapa había que programar e interactuar con los usuarios para darles algo a medida.

Cierre: Consistía en poner en producción el sistema y brindar la capacitación correspondiente a los usuarios.

2.1 GESTIÓN DE CRONOGRAMA

2.1.1 DEFINIR ACTIVIDADES

Para la elaboración del proyecto definiremos las actividades a seguir

- Investigación
- Definición y Alcance del Proyecto
- Definición de los Objetivos
 - General
 - Específicos
- Gestión de Cronograma - Definir Calendarización
- Gestión de Riesgos – Identificar los Riesgos
- Generar Especificación de Requerimientos del Sistema
- Arquitectura del Hardware
- Arquitectura del Software
- Producción
- Desarrollo
- Plan de Pruebas
- Testing
- Elaboración del Manual de Usuario

2.2 Calendarización

Para llevar a cabo este proyecto se detectaron a grandes rasgos las siguientes actividades que fueron separadas por tareas asignándoles un número; un nombre; una duración en días y las dependencias entre ellas, es decir qué tareas deben ser completadas antes de poder empezar otra. A continuación se detallan en el siguiente cuadro:

ACTIVIDADES	TIEMPO ESTIMADO/días
Investigación	2

Definición y Alcance del Proyecto	1
Definición de los Objetivos General y Específico	2
Definir Calendarización	15
Identificar los Riesgo	8
Generar Especificación de Requerimientos del Sistema	40
Arquitectura del Hardware	25
Arquitectura del Software	25
Desarrollo	95
Testing	15
Elaboración del Manual de Usuario	10

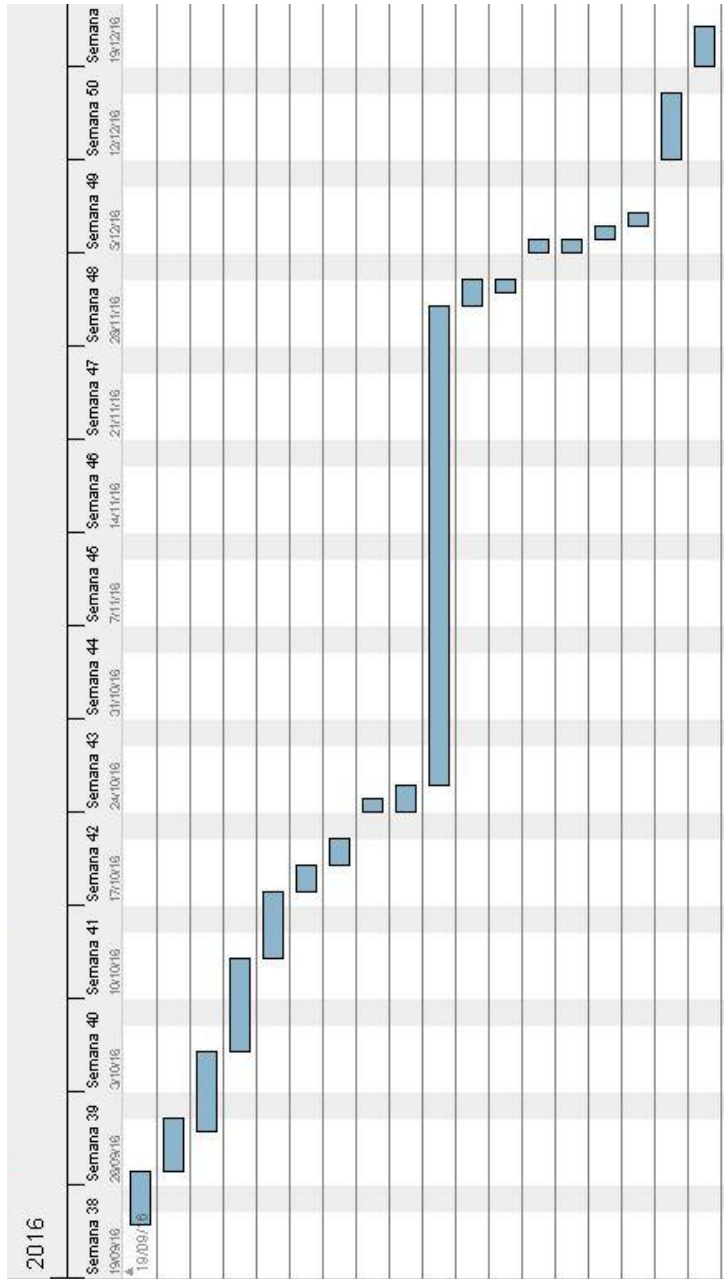
2.2.1 DIAGRAMA DE GANTT DEL PROYECTO

Para generar el diagrama de Gantt para la calendarización de actividades y tareas del proyecto utilizamos la herramienta Gantt Project

Calendarización de actividades

Una vez que se detectaron todas las tareas se procedió a armar un diagrama de Gantt, esto es una herramienta gráfica que muestra el tiempo de dedicación previsto para cada tarea y las dependencias entre ellas

Gantt project	Nombre		Fecha de ini...		Fecha de fin	
	● Investigación		23/09/16		26/09/16	
	● Análisis de Requerimientos		27/09/16		30/09/16	
	● Definición de Hitos y Entregas		30/09/16		5/10/16	
	● Definir la pila del producto		6/10/16		12/10/16	
	● Personalizar SCRUM		13/10/16		17/10/16	
	● Modelo del Dominio		18/10/16		19/10/16	
	● Base de Datos		20/10/16		21/10/16	
	● Arquitectura del Software		24/10/16		24/10/16	
	● Prototipos		24/10/16		25/10/16	
	● Desarrollo del sistema Web		26/10/16		30/11/16	
	● Casos de Prueba		1/12/16		2/12/16	
	● Pruebas del Software		2/12/16		2/12/16	
	● Resultados de las pruebas		5/12/16		5/12/16	
	● Verificación & Validación		5/12/16		5/12/16	
	● Pruebas de Calidad		6/12/16		6/12/16	
	● Puesta en Marcha		7/12/16		7/12/16	
	● Manuales de Usuario		12/12/16		16/12/16	
	● Capacitación de Usuarios		19/12/16		21/12/16	



Nro	Actividades	Dependencias	Inicio	Fin
1	Investigación		23/9/16	26/09/16
2	Análisis de Requerimientos	T1	27/09/16	30/09/16
3	Definición de Hitos y Entregas	T2	30/09/16	05/10/16
4	Definir la pila del producto	T1,T2,T3	06/10/16	12/10/16
5	Personalizar SCRUM	T2	13/10/16	17/10/16
6	Modelo del Dominio	T1, T2	18/10/16	19/10/16
7	Base de Datos	T1, T2	20/10/16	21/10/16
8	Arquitectura del Software	T6, T7	24/10/16	24/06/14
9	Prototipos	T4, T2	24/10/16	25/10/16
10	Desarrollo del sistema Web	T9, T2	26/10/16	30/11/16
11	Casos de Prueba	T2, T10	1/12/16	02/12/16
12	Pruebas del Software	T10, T11	02/12/16	02/12/16
13	Resultados de las pruebas	T11, T2	05/12/16	05/12/16
14	Verificación & Validación	T11, T12, T13	05/12/16	05/12/16
15	Pruebas de Calidad	T11, T12	06/12/16	06/12/16
16	Puesta en Marcha	T15, T14	07/12/16	07/12/16
17	Manuales de Usuario	T14, T13	12/12/16	16/12/16
18	Capacitación de Usuarios	T16, T17	19/12/16	21/12/16

FASE DE INICIACIÓN

3.1 Hitos y entregas

Hito N° 1:

- Actividades:
 - Investigación.
 - Definición de Requerimientos.
 - Planificación.
 - Selección de personal.
- Entrega:
 - Planificación de los Sprint (duración y fechas).
 - Selección del Personal (equipos) y asignación de roles.
 - Product Backlog (Especificación de los Requerimientos).

Hito N° 2:

- Actividades / Entregables:
 - Personalizar SCRUM.
 - Diseño de la Base de Datos
 - Modelo del Dominio.
 - Arquitectura del Software.
 - Diagrama de Bloques.
 - Prototipos.

Hito N° 3:

- Actividades:
 - Programación del Sistema Web.
 - Casos de Prueba
- Entrega:
 - Prototipos Funcionales.
 - Software Instalado.

Hito N° 4:

- Actividades:
 - Pruebas

- Entrega:
 - Resultados de las Pruebas.

Hito N° 5:

- Actividades:
 - Puesta en Marcha.
 - Confección de Manuales de Usuario.
- Entrega:
 - Manuales de Usuario.
 - Capacitación de Usuarios.

Hito N° 6:

- Actividades:
 - Evaluación del Uso y Test en producción.
- Entregas:
 - Evaluación Final / Conformidad del cliente.

3.2 PERSONALIZANDO SCRUM:

A continuación se detallan los tipos de reuniones, duraciones y fechas de los Sprint.

3.2.1 DAILY SCRUMS:

Se utilizarán 15 minutos de cada día para reunirse (reunión informal o Daily Scrum). Dichas reuniones tendrán las siguientes pautas:

- La reunión comienza puntualmente a las 9.
- Hay castigos -acordados por el equipo- para quien llegue tarde.
- Duración fija de 15 minutos.
- Misma ubicación y a la misma hora todos los días.
- Todos deben mantenerse de pie (reunión corta).
- Todos son bienvenidos, pero solo los responsables pueden hablar.
- Durante la reunión, cada miembro del equipo contestará cuatro preguntas:
 - 1) ¿Qué has hecho desde ayer?
 - 2) ¿Has tenido algún problema que te haya impedido alcanzar tu objetivo?

- 3) ¿Qué es lo que estás planeando hacer hoy?
- 4) ¿Tienes algún problema que te impida continuar con tu objetivo para hoy?

3.2.2 REUNIONES DE PLANIFICACIÓN DE LOS SPRINT (Sprint Planning):

Estas reuniones se llevarán a cabo cuando inicie cada Sprint. A estas reuniones deberán concurrir obligatoriamente todos los miembros (Product Owner, Scrum Master, Equipos, Stakeholders en caso de requerirse y cualquier persona que fuese necesaria).

Estas reuniones tienen como objetivo definir cuáles serán las próximas tareas a desarrollar en el Sprint. El lugar destinado a reuniones puede variar, el scrum master será el encargado de avisar el lugar y horario de la reunión a los integrantes del equipo.

Los sprints tendrán como máxima duración 8 horas, con un break en el medio de 30 minutos a 1 hora.

En estas reuniones se tratarán temas como:

- Selección del trabajo que se hará.
- Preparar, con el equipo completo, el Sprint Backlog que detalla el tiempo que tomará hacer el trabajo y las tareas a desarrollar.
- Identificar y comunicar cuánto del trabajo es probable que se realice durante el actual Sprint.
- Planificaciones futuras, Exposiciones, Requerimientos.

Las fechas aproximadas de cada Sprint Planning se detallan a continuación:

SPRINT	INICIA	HORA	LUGAR	DURACION
1 – Inicial	23/09/2016	09:00	A definir	6 horas
2	17/10/2016	09:00	A definir	6 horas
3	26/10/2016	09:00	A definir	6 horas
4	04/11/2016	09:00	A definir	6 horas
5	28/11/2016	09:00	A definir	6 horas
6	04/12/2016	A convenir	A definir	6 horas
7 – Final	12/12/2016	A convenir	A definir	6 horas

1 – TABLA DE SPRINTS

SPRINT 1

Tomará gran parte del Hito 1 y durará 4 semanas. Fin: 14/10/2016

Investigación, y la <especificación de requisitos del sistema (ERS.)

SPRINT 2

El resto del Hito 1 con una duración de 2 semanas. Fin: 25/10/2016

Registro de Interesados, Definición de Hitos y Entregas, Selección del Personal, Product Backlog.

SPRINT 3

Tendrá una duración de 1 semana, finaliza el 03/11/2016 y las tareas a desarrollar son: Modelo del Dominio, Diseño de Base de Datos, Arquitectura Software, Diagrama de Despliegue, Prototipos.

SPRINT 4

Duración de 3 semanas, concluye el 25/11/2016 y abarca: Programación del sitio web, Instalación del Software, Casos de Prueba.

SPRINT 5

Duración de 1 semana, finaliza el 03/12/2016 y las tareas son: Pruebas de Software, Resultados de las pruebas, V&V, Pruebas de Calidad.

SPRINT 6

Duración 1 semana. Abarca: Confección de Manuales y Puesta en Marcha

SPRINT 7

Es el sprint final y no debe exceder 1 semana. Las tareas que abarca son:

Capacitación de los usuarios, Evaluación y Conformidad del cliente y Pruebas en entorno de producción.

3.3 Pila del Producto

A la pila del producto la define el Product Owner (dueño del producto) e incluye todas las actividades que hay que hacer para finalizar el proyecto, estimaciones, una breve descripción de la funcionalidad deseada y la prioridad.

US01	Crear un nuevo evento					
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional	
Descripción	El Administrador registra un nuevo evento indicando si el mismo es público o interno, el sistema deberá mostrar la opción pública por defecto permitiendo al usuario cambiarla. Además se deberá incluir siguientes campos obligatorios:					

	evento organizador fecha estado color apellidoNombre email teléfono objetivos destinatarios asistentes catering proyector log entrada salida tipo
--	---

US02	Modificar evento				
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional
Descripción	El Administrador podrá modificar la información relacionada a un de un evento registrado en el sistema				

US03	Cancelar evento				
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional
Descripción	El Administrador podrá cambiar el estado de un evento activo a cancelado				

US04	Listar los eventos semanales				
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional
Descripción	El Administrador podrá listar los todos eventos de la semana en curso que se encuentren registrados en el sistema. Mostrando los siguientes datos: Evento Organizador Horario Fecha Lugar Opciones (ver detalle, editar, cancelar, eliminar)				

US05	Listar los eventos del día					
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional	
Descripción	El Administrador podrá listar los todos eventos del día en curso que se encuentren registrados en el sistema. Mostrando los siguientes datos: Evento Organizador Horario Fecha Lugar Opciones (ver detalle, editar, cancelar, eliminar)					

US06	Listar los eventos mensuales					
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional	
Descripción	El Administrador podrá listar los todos eventos del mes en curso que se encuentren registrados en el sistema. Mostrando los siguientes datos: Evento Organizador Horario Fecha Lugar Opciones (ver detalle, editar, cancelar, eliminar)					

US07	Listar todos los eventos					
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional	
Descripción	El Administrador podrá listar los todos eventos registrados en el sistema. Tanto públicos como privados y cancelados. Mostrando los siguientes datos: Evento Organizador Horario Fecha Lugar Opciones (ver detalle, editar, cancelar, eliminar)					

US08	Publicar un evento					
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional	
Descripción	El Administrador podrá publicar un evento para ser mostrado en la pantalla de información al público.					

US09	Crear una nuevo espacio físico					
Prioridad	Alta/Esencial	x	Media/Deseado		Baja/ Opcional	
Descripción	Administrador podrá crear un nuevo espacio físico con la siguiente informacion:					

	Espacio Físico Nombre Ubicación Capacidad Recursos Estado
--	--

US10	Modificar un espacio físico					
Prioridad	Alta/Esencial	<input checked="" type="checkbox"/>	Media/Deseado	<input type="checkbox"/>	Baja/ Opcional	<input type="checkbox"/>
Descripción	El Administrador podrá modificar la información asociada a un espacio físico					

US11	Mostrar calendario					
Prioridad	Alta/Esencial	<input checked="" type="checkbox"/>	Media/Deseado	<input type="checkbox"/>	Baja/ Opcional	<input type="checkbox"/>
Descripción	El Administrador podrá consultar los eventos registrados a través de un calendario donde se podrán filtrar los eventos por diarios, semanales y mensuales					

US12	Diferenciar tipos de eventos en el calendario					
Prioridad	Alta/Esencial	<input checked="" type="checkbox"/>	Media/Deseado	<input type="checkbox"/>	Baja/ Opcional	<input type="checkbox"/>
Descripción	El Administrador podrá visualizar los eventos a través del calendario, donde se podrán diferenciar por colores diferentes dependiendo si el evento es público, interno, o si se encuentra cancelado.					

US13	Exportar eventos en formato excel					
Prioridad	Alta/Esencial	<input checked="" type="checkbox"/>	Media/Deseado	<input type="checkbox"/>	Baja/ Opcional	<input type="checkbox"/>
Descripción	El Administrador podrá exportar los eventos registrados filtrando el mes que desee					

US14	Login de usuario					
Prioridad	Alta/Esencial	<input checked="" type="checkbox"/>	Media/Deseado	<input type="checkbox"/>	Baja/ Opcional	<input type="checkbox"/>
Descripción	El usuario podrá ingresar al sistema con un mail y una contraseña. El sistema deberá validar los datos ingresados para confirmar que el usuario es un usuario registrado, caso contrario se deberá mostrar un mensaje informativo solicitando que se ingresen los datos correctos.					

3.4 MODELO DE DATOS

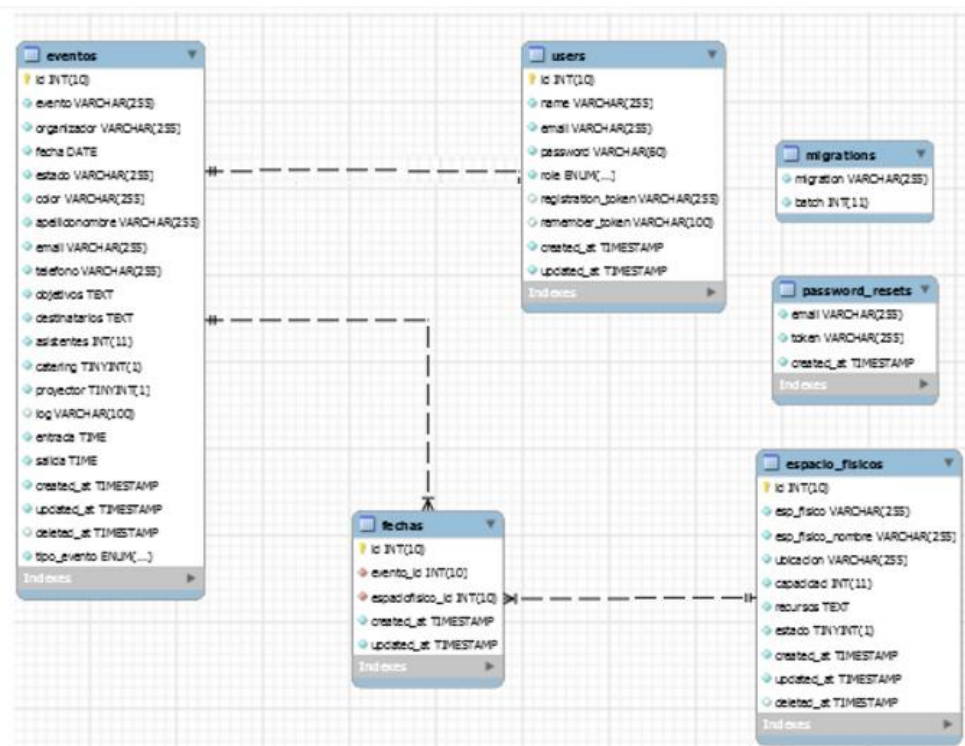


Figura 17 - Modelo de Datos - Diagrama Relacional

3.5 MODELO DE DATOS

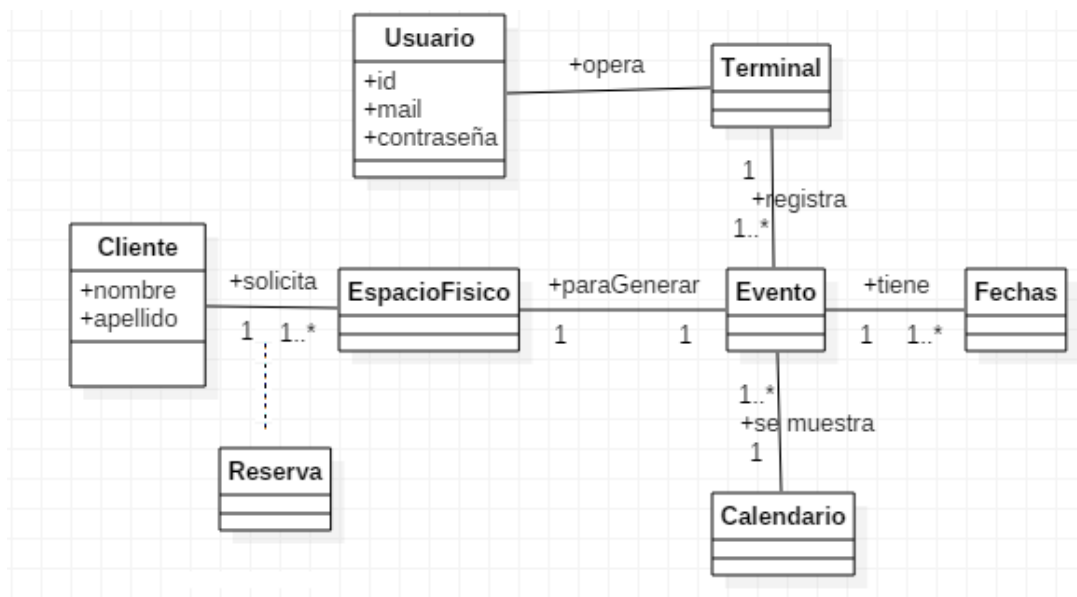


Figura 17 - Modelo de Dominio

FASE DE PRODUCCIÓN

4.1 LENGUAJES DE PROGRAMACIÓN UTILIZADOS

4.1.1 PHP



Figura 18 – Logo PHP

PHP es un lenguaje interpretado de propósito general ampliamente usado y que está diseñado al usar especialmente para desarrollo Web y puede ser incrustado dentro de código HTML. Generalmente se ejecuta en un servidor Web, tomando el código en PHP como su entrada y creando páginas Web como salida. Puede ser desplegado en la mayoría de los servidores Web y en casi todos los sistemas operativos y plataformas sin costo alguno. PHP se encuentra instalado en más de 20 millones de aplicaciones web y en un millón de servidores, el número de sitios en PHP ha compartido algo de su preponderante sitio con otros nuevos lenguajes no tan poderosos desde agosto de 2005. Este mismo sitio Web de Wikipedia está desarrollado en PHP. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor Web. La versión más reciente de PHP es la 5.3.1 (for Windows) del 19 de noviembre de 2009. Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX (y de ese tipo, como Linux o Mac OS X) y Windows, y puede interactuar con los servidores de Web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

¿Cómo funciona?

Cuando el cliente hace una petición al servidor para que le envíe una página Web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos). El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. Mediante extensiones es también posible la generación de archivos PDF, Flash, así como imágenes en diferentes formatos. Por otra parte, el sistema de gestión de bases de datos que contendrá toda la información necesaria para ser mostrada en la aplicación es MySQL. Una vez decidido esto, necesitamos un servidor Web que nos permita ejecutar el código en PHP a la vez que nos permita acceder a la base de datos MySQL. El servidor Web utilizado es Apache. El protocolo utilizado para la transmisión de datos entre redes de computadoras es el protocolo TCP/IP. Además, el protocolo http será el que nos permita el acceso a la aplicación web. En conclusión decidí usar PHP como lenguaje debido a que presenta grandes ventajas:

- Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos.

4.1.2 CSS



Figura 19 – Logo CSS

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento. CSS funciona a base de reglas, es decir, declaraciones sobre el estilo de uno o más elementos. Las hojas de estilo están compuestas por una o más de esas reglas aplicadas a un documento HTML o XML. La regla tiene dos partes: un selector y la declaración. A su vez la declaración está compuesta por una propiedad y el valor que se le asigne.



Figura 20 – Logo Bootstrap

Bootstrap, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como “*responsive design*” o diseño adaptativo.

El beneficio de usar responsive design en un sitio web, es principalmente que el sitio web se adapta automáticamente al dispositivo desde donde se acceda.

Aun ofreciendo todas las posibilidades que ofrece Bootstrap a la hora de crear interfaces web, los diseños creados con Bootstrap son simples, limpios e intuitivos, esto les da agilidad a la hora de cargar y al adaptarse a otros dispositivos. El Framework trae varios elementos con estilos predefinidos fáciles de configurar: Botones, Menús desplegables, Formularios incluyendo todos sus elementos e integración jQuery para ofrecer ventanas y tooltips dinámicos.

Bootstrap tiene un soporte relativamente incompleto para [HTML5](#) y [CSS 3](#), pero es compatible con la mayoría de los navegadores web.

4.1.4 MySQL



Figura 21 - Logo MySQL

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL. Desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C. Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

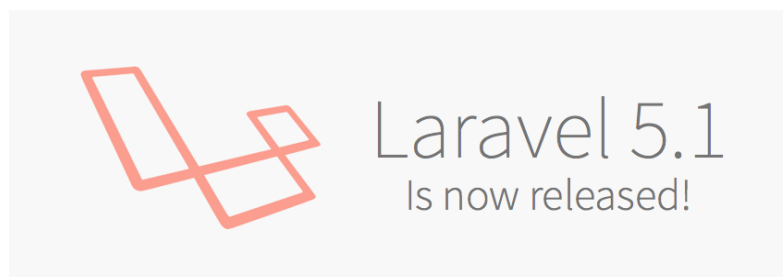
4.1.5 APACHE



Logo de Servidor Apache

El servidor Apache es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1 (RFC 2616) y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA Http 1.3, pero más tarde fue reescrito por Completo. Su nombre se debe a que originalmente Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, a patchy server (un servidor parcheado). El servidor Apache se desarrolla dentro del proyecto HTTP Server (httpd) de la Apache Software Foundation. Apache presenta entre otras cosas mensajes de error altamente configurables, bases de datos de Autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración. En la actualidad (2005), Apache es el servidor HTTP más usado, siendo el servidor HTTP del 68% de los sitios Web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft

4.1.6 LARAVEL 5.1



Logo Laravel 5.1

Este proyecto fue desarrollado con Laravel 5.1, laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti". Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.

Laravel tiene como objetivo ser un framework que permita el uso de una sintaxis elegante y expresiva para crear código de forma sencilla y permitiendo multitud de funcionalidades. Intenta aprovechar lo mejor de otros frameworks y aprovechar las características de las últimas versiones de PHP.2

Gran parte de Laravel está formado por dependencias, especialmente de Symfony, esto implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias.

Características

- Sistema de ruteo, también RESTful
- Blade, Motor de plantillas
- Peticiones Fluent
- Eloquent ORM
- Basado en Composer
- Soporte para el caché
- Soporte para MVC
- Usa componentes de Symfony

Plantillas

Las plantillas en Blade son archivos de texto plano que contiene todo el HTML de la página con etiquetas que representan elementos o zonas a incluir en la plantilla, o vistas parciales como se conocen en otros frameworks PHP. Sin embargo, en Blade estos elementos incrustados se organizan en un sólo archivo. Esta es una idea muy interesante de Laravel que mejora la organización de las vistas y su rendimiento. Sobre todo cuando las vistas pueden llegar a ser muy complejas incluso con elementos anidados. En el render de una Vista completa en Laravel se usan dos archivos: la plantilla definiendo el HTML global y las zonas a incluir. Un sólo archivo, la Vista, con los elementos (partial views).

Influencia

La influencia de Laravel ha crecido rápidamente desde su lanzamiento. En la comunidad de desarrolladores es considerado como alternativa sencilla de usar pero que tiene todas las funcionalidades que debe tener un framework. Ha sido descargado más de 320.000 veces, y se espera que supere en popularidad a otros frameworks ya establecidos más antiguos.

5 PLAN DE PRUEBAS

5.1 MARCO TEÓRICO DE LAS PRUEBAS DE SISTEMA

Las pruebas de software son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada. Es una actividad más en el proceso de control de calidad.

Las pruebas son básicamente un conjunto de actividades dentro del desarrollo de software. Dependiendo del tipo de pruebas, estas actividades podrán ser implementadas en cualquier momento de dicho proceso de desarrollo. Existen distintos modelos de desarrollo de software, así como modelos de pruebas. A cada uno corresponde un nivel distinto de involucramiento en las actividades de desarrollo.

5.2 PLAN DE PRUEBAS DE SISTEMA

5.2.1 OBJETIVO DE LA PRUEBA

Asegurar la apropiada navegación dentro del sistema, ingreso de datos, procesamiento y recuperación.

5.2.2 DESCRIPCIÓN DE LA PRUEBA

Las pruebas del sistema deben enfocarse en requisitos que puedan ser tomados directamente de casos de uso y reglas y funciones de negocios. El objetivo de estas pruebas es verificar el ingreso, procesamiento y recuperación apropiado de datos, y la implementación apropiada de las reglas de negocios. Este tipo de pruebas se basan en técnicas de caja negra, esto es, verificar el sistema (y sus procesos internos), la interacción con las aplicaciones que lo usan vía GUI y analizar las salidas o resultados.

En esta prueba se determina qué pruebas de Sistema (usabilidad, volumen, desempeño, etc.) asegurarán que la aplicación alcanzará sus objetivos de negocio.

La prueba de Sistema incluye:

- Prueba de Funcionalidad

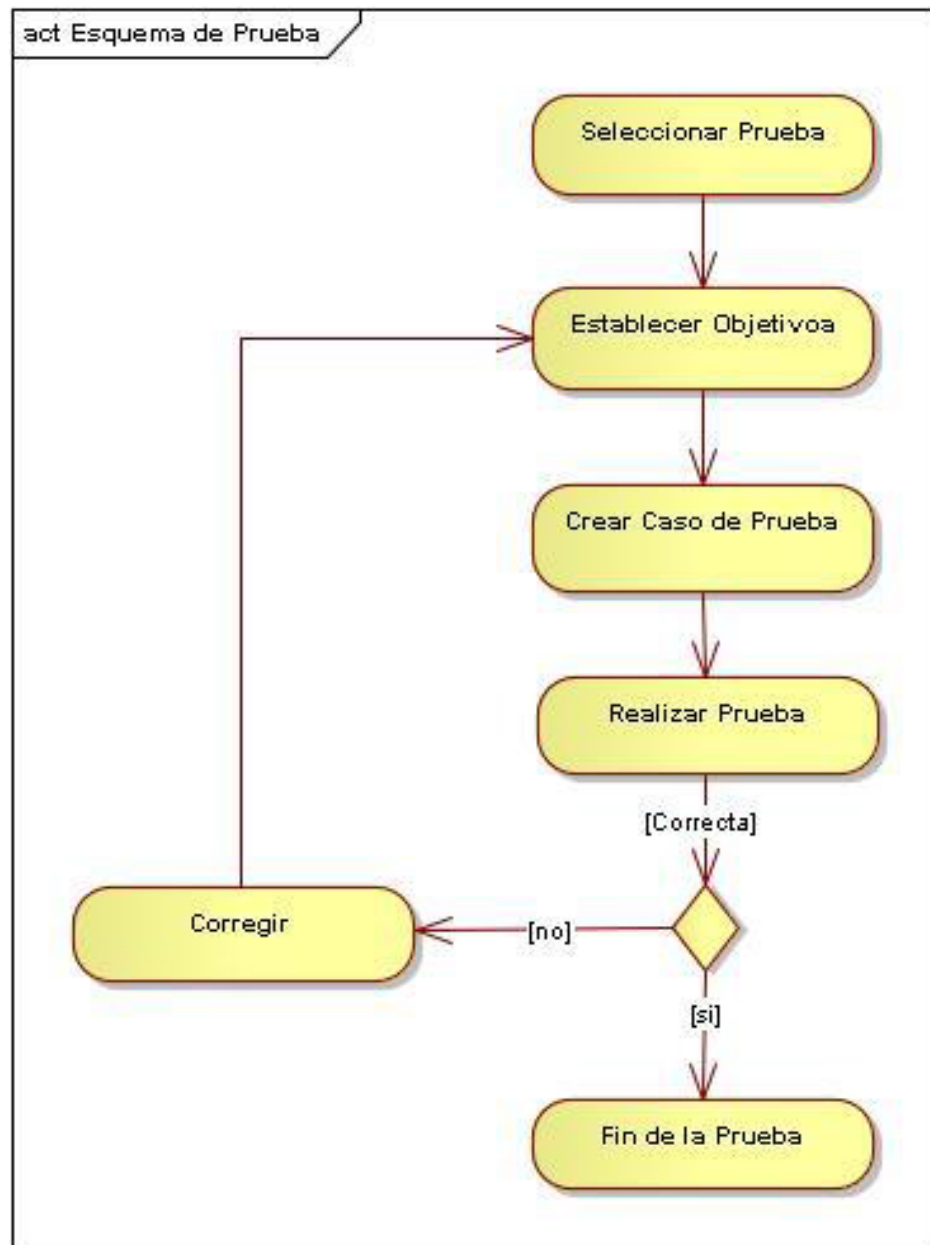
- Prueba de Usabilidad
- Prueba de Performance
- Prueba de Documentación y Procedimientos
- Prueba de Seguridad y Controles
- Prueba de Volumen
- Prueba de Esfuerzo (Stress)
- Prueba de múltiples sitios

Se desarrollarán Casos de Prueba para verificar la documentación, procedimientos y desempeño que no han sido probados durante la prueba unitaria y de integración.

La prueba de sistema es compleja porque intenta validar un número de características al mismo tiempo, a diferencia de otras pruebas que sólo se centran en uno o dos aspectos del sistema al mismo tiempo.

Para todos los Casos de Prueba, verificar la documentación de la Especificación de Requisitos de Software (Anexo B)

5.2.3 ESQUEMA DE LAS PRUEBAS



5.3 PASOS DEL PLAN DE PRUEBAS

5.3.1.CONDICIONES

Se establecen que cada Caso de Uso propuesto en el apartado 3.2.3, será un Caso de Prueba.

Los participantes o Testing de las Pruebas serán elegidos al azar, a cada uno se facilitará el link para realizar la Prueba.

Los Testing deberán completar la Plantilla propuesta para el seguimiento de la Prueba.

Se deberán utilizar distintos dispositivos para la ejecución de las Pruebas.

Al fin de cada Prueba el Equipo evaluará las mismas, estableciendo los pasos a seguir para las correcciones o mejoras sugeridas.

5.3.2 PLANTILLA PARA LAS PRUEBAS

Ver Anexo C – Plantilla de Prueba.

Las Secciones de la Plantilla están diseñadas para obtener los resultados de las distintas Pruebas.

5.4 REPORTES

5.4.1 INTRODUCCIÓN

En esta fase de transición y luego de realizar las Pruebas correspondientes del sistema en un periodo determinado, podemos plantear los reportes de Ventas que tendrá el sistema.

Estos reportes nos permitirán ver la evolución de las ventas de los comercios asociados y definir nuevas estrategias para aumentarlas.

5.2.2 REPORTES DE EVENTOS

Para visualizar los reportes de eventos vamos a acceder al menú exportar:

Se presentará la siguiente pantalla:

Figura 23 - Reportes - Inicio

Además podemos filtrar por fecha

Figura 27 - Reportes - Filtrar

Y podemos imprimirlo

Figura 28 - Reportes - Imprimir

CONCLUSIONES

6 CONCLUSIONES

De acuerdo a las pautas establecidas en la Planificación del Proyecto, puedo decir que se ha logrado el objetivo perseguido, la creación de la Aplicación Web de agenda de reservas está en funcionamiento y puede ser accedida sin inconvenientes.

La documentación provista en este proyecto puede ser interpretada de manera clara y sencilla a través de una redacción en este mismo sentido.

Las pruebas propuestas en el Plan de Pruebas se llevaron a cabo sin ningún inconveniente y la puesta en marcha del sistema es totalmente exitosa.

A lo largo del desarrollo de este proyecto tuve la oportunidad de adquirir experiencia y de afianzar los conocimientos adquiridos en estos años de estudio, su desarrollo ha integrado los conceptos que se impartieron en las distintas materias de la carrera, consiguiendo un producto de calidad basado en las buenas prácticas adquiridas.

ANEXOS

7 ANEXOS

7.1 ANEXO A – ACTA DE CONSTITUCIÓN DEL PROYECTO

7.1.1 INFORMACIÓN GENERAL

NOMBRE DEL PROYECTO	PATROCINADOR	FECHA DE PRESENTACIÓN
Agenda de Reservas	Ayelén Fernández	12/12/2016

HISTORIAL DE VERSIONES

VERSION	PRESENTADO POR	FECHA
1.0	Ayelén Fernández	20/11/2016

VERSION	PRESENTADO POR	FECHA
1.1	Ayelén Fernández	30/11/2016

VERSION	PRESENTADO POR	FECHA
---------	----------------	-------

2.0	Ayelén Fernández	05/11/2016
-----	------------------	------------

PROPIETARIO	ESTADO	FECHA
Ayelén Fernández	autorizado	05/11/2016

7.1.2 ANTECEDENTES

El CIIDEPT es un centro de innovación tecnológica educativa, que ofrece capacitaciones, charlas, seminarios y eventos de diversas temáticas educativas, así como también ofrece el uso de espacios físicos a diferentes organismos públicos, tanto de la provincia como nación, para la realización de diferentes actividades orientadas al sector educativo.

El objetivo del sistema es facilitar la asignación y gestión de los eventos tanto públicos como internos del establecimiento, con una mejor utilización y administración de los recursos disponibles.

El servicio consiste en brindar una plataforma online simple, práctica y sin costo adicional que permite a nuestros usuarios asignar los espacios y recursos disponibles de acuerdo a los diferentes eventos que se realiza a diario en la institución.

7.1.3 JUSTIFICACIÓN DEL PROYECTO

El sitio web de agenda de reservas concentra en una sola página la posibilidad de gestionar los eventos realizados en la institución de manera rápida y sencilla.

7.1.4 IDENTIFICACIÓN DE LOS INTERESADOS

INTERESADO	SECTOR
CLIENTE	Sector administrativo del ciidept
MINISTRO DE EDUCACIÓN	Dr. Juan Pablo Lichtmajer

DOCENTES	Docentes de la provincia de Tucumán
DOCENTES DE LA CÁTEDRA DE HABILITACIÓN PROFESIONAL	Docentes de la Cátedra de Habilitación Profesional de la Carrera de Ingeniería en Sistemas de Información de la UTN - FRT.
ALUMNO DESARROLLADOR DEL PROYECTO	Ayelén Fernández

7.1.5 REQUISITOS QUE SATISFACEN LAS NECESIDADES, DESEOS Y EXPECTATIVAS DE LOS INTERESADOS

7.1.5.1 FUNCIONALES

El sistema de agenda de reservas debe mostrar todos los eventos registrados, con la información referida a cada uno de ellos.

El sistema deberá mostrar por pantallas todos los eventos públicos que se realizarán en el día, con el objetivo de brindar información rápida a los docentes, acerca de las actividades que se realizan en la institución.

7.1.5.2 TÉCNICOS

El sistema debe facilitar a los administrativos la búsqueda rápida y efectiva de los diferentes eventos registrados, con el motivo de brindar información válida y concisa ante la consulta de los interesados.

La actualización y/o cancelación de los eventos debe ser muy sencilla, los diferentes colores utilizados en el calendario de eventos, permitirá a los usuarios diferenciar los estados y tipos de eventos de una manera rápida y clara a través de una leyenda, la cual indica las referencias para cada color.

7.1.6 OBJETIVOS DEL PROYECTO

El sistema web de agenda de reservas, debe estar listo para el 07 de Diciembre del 2016

7.1.7 GERENTE DEL PROYECTO Y SU NIVEL DE AUTORIDAD

NOMBRE DEL GERENTE DEL PROYECTO Ayelén Fernández

RESPONSABILIDADES PRINCIPALES

Elaborar el Plan del Proyecto y asegurar que se cumpla lo establecido.

Llevar a cabo labores de seguimiento y control: Reportar periódicamente la situación del proyecto, actualizar periódicamente el Plan del Proyecto.

Tomar decisiones sobre aspectos que tengan impacto en el alcance, costo o plazo comprometidos.

Asegurar la integridad y calidad de los entregables generados en el proyecto.

ATRIBUCIONES PRINCIPALES

Tendrá la suficiente autoridad y atribución como para resolver los conflictos que puedan poner en riesgo los objetivos y metas del proyecto.

7.1.8 RESTRICCIONES DE LA ORGANIZACIÓN

- El presupuesto no podrá exceder al 10% del monto aprobado.
- Cualquier adicional al presupuesto inicial deberá ser aprobado por la Gerencia General para lo cual se presentará un documento con el sustento de la ampliación.
- El costo de mantenimiento no deberá exceder al 10% del presupuesto del proyecto.
- Debe garantizarse la integridad y confidencialidad de la información de la empresa que puede verse afectada a través del acceso del público en general.
- No se hará ninguna inversión en equipos ni infraestructura tecnológica para el desarrollo del proyecto ni durante la operación del producto.
- El sistema Web debe ser probado durante un mes por los usuarios que utilizaran el sistema.

7.1.9 SUPUESTOS (ASUNCIONES) DE LA ORGANIZACIÓN

El desarrollo del proyecto deberá estar alineado con las pautas del PMBOK. Se mantendrá un repositorio para los textos e imágenes utilizados en el proyecto en el momento que sea requerido de acuerdo al Plan del proyecto. Las personas responsables de aprobar los entregables se sujetarán a los plazos establecidos en el Plan del Proyecto.

7.1.10 PRESUPUESTO

Costo de Implementación. Requerimientos de material informático como software y hardware adecuados a las funcionalidades exigidas al sitio y recursos humanos capaces de llevar adelante el proyecto, esto puede implicar contratación de personal, capacitación de personal o gastos por externalización. Para este Proyecto no aplica.

Costos de Desarrollo. Cada etapa considerada en el modelo conceptual del sitio estará asociada a costos de desarrollo. En este sentido es adecuado revisar la posibilidad de implementar el sitio mediante un sistema escalable, en que el costo de desarrollo pueda dividirse en etapas; de esta manera se evitará un impacto fuerte en el presupuesto. Para este Proyecto no aplica.

Costos de Mantenimiento y Actualización. Implica gastos en personal, rediseño y ajuste de procedimientos y actualización de la información. En este ítem se deben considerar los gastos informáticos relacionados, como el alojamiento del Sitio Web (hosting) y la conectividad, entre otros. Los costos de Hosting en un servidor dedicado es de aproximadamente 135 a 140 dólares por mes.

7.1.10.2 BENEFICIOS ESPERADOS

- Registro de eventos con fines educativos realizados en la Provincia de Tucumán.
- Eficaz administración y asignación de los espacios físicos y recursos disponibles.
- Brindar información rápida y precisa a los docentes o solicitantes.

7.1.11 PROGRAMA DE HITOS Y ENTREGABLES DEL PROYECTO

HITO	ACTIVIDADES	ENTREGABLE
Hito N° 1	<ul style="list-style-type: none"> o Investigación. o Definición de Requerimientos. o Planificación. o Selección de personal. 	<ul style="list-style-type: none"> o Planificación de los Sprint (duración y fechas). o Selección del Personal (equipos) y asignación de roles. o Product Backlog (Especificación de los Requerimientos).
Hito N° 2	<ul style="list-style-type: none"> o Personalizar SCRUM. o Diseño de la Base de Datos o Modelo del Dominio. o Arquitectura del Software. o Diagrama de Bloques. o Prototipos. 	<ul style="list-style-type: none"> o Base de Datos o Modelo del Dominio. o Arquitectura del Software. o Diagrama de Bloques. o Prototipos.
Hito N° 3	<ul style="list-style-type: none"> o Programación del Sistema Web. o Casos de Prueba 	<ul style="list-style-type: none"> o Prototipos Funcionales. o Software Instalado.
Hito N° 4	<ul style="list-style-type: none"> o Pruebas 	<ul style="list-style-type: none"> o Resultados de las Pruebas.
Hito N° 5	<ul style="list-style-type: none"> o Puesta en Marcha. o Confección de Manuales de Usuario. 	<ul style="list-style-type: none"> o Manuales de Usuario. o Capacitación de Usuarios.
Hito N° 6	<ul style="list-style-type: none"> o Evaluación del Uso y Test en producción. 	<ul style="list-style-type: none"> o Evaluación Final / Conformidad del cliente.

Tabla 4 - Hitos y Entregables del Proyecto

7.1.12 INTEGRANTES DEL EQUIPO DEL PROYECTO, ROLES.

NOMBRE	ROL
Ayelén Fernández	Gerente General
Ayelén Fernández	Analista Funcional
Ayelén Fernández	Director de Proyecto
Facundo Ruíz	Tester

Tabla 5 - Integrantes del equipo del proyecto

7.2 ANEXO B – ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

7.2.1 INTRODUCCIÓN

Esta documentación responde al estándar IEEE 830 para la Especificación de Requisitos de Software y es la conclusión de la primera etapa de Análisis de Requisitos.

CIIDEPT es un centro de Innovación Tecnológica Educativa que depende del Ministerio de Educación de la Provincia de Tucumán. En un principio estaba destinado a la realización de eventos de índole educativa, en los últimos años se comenzó a utilizar los espacios físicos para diferentes propósitos con fines educativos, como ser charlas, seminarios, capacitaciones, reuniones, entre otros.

El objetivo del sistema web es facilitar a los usuarios la tarea de registrar dichos eventos en una agenda, asignando los recursos necesarios de manera fácil y efectiva, de acuerdo al pedido del solicitante el pedido. Por parte beneficiamos a los docentes que concurren a los diferentes eventos, ya que cuentan con pantallas informativas, donde se muestran los eventos del día para una mayor comodidad a la hora de consultar horarios y aulas donde se dicta dicho evento.

El servicio consiste en brindar una plataforma online simple, práctica y sin costo adicional

7.2.2 PROPOSITO

El Propósito fundamental de esta documentación es especificar en forma clara, sencilla y concisa los requisitos necesarios para que el nuevo sistema funcione correctamente.

Este documento está preparado de forma tal, que podrán leerlo todos los Interesados identificados en el Acta de Constitución del Proyecto (Anexo A).

7.2.3 ALCANCE

El Alcance del Proyecto es abordar las operaciones comerciales del sitio WEB desde el punto de vista del usuario Cliente:

- Iniciar Sesión de Usuario
- Registrar Evento
- Consultar Eventos
- Registrar Espacio físico
- Consultar Espacios físicos
- Generar reportes de Eventos
- Visor de eventos en pantalla pública
- Visualizar eventos en calendario
- Cerrar sesión de usuario

7.2.4 OBJETIVOS

Los Objetivos perseguidos por el Proyecto son:

- Disponer de la información de manera eficiente, de una forma sencilla e intuitiva para el usuario que accede al sistema
- Mantener actualizada la información de los eventos registrados
- Establecer un vínculo con el Cliente para poder implementar mejoras continuas
- Optimizar las operaciones que llevan a cabo los usuarios Clientes como ser Iniciar Sesión de Usuario

- Mediante un Nombre de Usuario y una Contraseña
- Permitir la Recuperación de Usuario/ Contraseña
- Manteniendo la información actualizadas sobre los avances en el sistema como ser nuevas funcionalidades

7.3 DEFINICIONES, ACRÓNIMOS Y ABREVIATURAS

7.3.1 DEFINICIONES

Sistema Web: se denomina así a aquellas aplicaciones que los usuarios pueden utilizar accediendo a un servidor web a través de Internet o de una intranet mediante un navegador.

Motor de base de datos MySQL: es un sistema de gestión de base de datos relacional. Sirve de interfaz entre la base de datos (es decir los datos almacenados), el usuario y las aplicaciones que la utilizan.

Navegador de Internet (por ejemplo: Google Chrome, Internet Explorer, Mozilla, Opera y otros): es una aplicación software que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML (lenguaje de programación para diseño de las páginas web), desde servidores web de todo el mundo a través de Internet. Esta red de documentos es denominada World Wide Web (WWW). Cualquier navegador actual permite mostrar o ejecutar gráficos, secuencias de vídeo, sonido, animaciones y programas diversos además del texto y los hipervínculos o enlaces.

Interfaces: es una herramienta gráfica que permite la interacción del usuario con la aplicación.

Web Services: un servicio web (en inglés Web service) es un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet.

Módulos: conjunto de aplicaciones que desarrolladas para acciones específicas.

User Stories: es una secuencia de acciones que realiza un actor (usuario) para concretar un determinado proceso. Por ejemplo las acciones que se realizan para la creación de un Evento.

Formatos de Archivos .pdf y .txt: son extensiones de archivos referidos a los formatos específicos de texto.

Formatos de Archivos xls: son extensiones de archivos referidos a los formatos específicos de hoja de cálculo.

Lenguaje PHP: es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. PHP es un acrónimo recursivo que significa PHP Hypertext Pre-processor.

SCRUM: SCRUM es un modelo de referencia que define un conjunto de prácticas y roles, y que puede tomarse como punto de partida para definir el proceso de desarrollo que se ejecutará durante un proyecto.

Páginas Dinámicas: son aquellas páginas de Internet que interactúan con una Base de Datos.

7.3.1.1 ACRONIMOS

ERS: Especificación de Requisitos de Software. Es el documento que se entrega al final de la etapa del análisis para la validación de los requisitos encontrados por los Analistas Funcionales.

IEEE: Instituto de Ingenieros en Electricidad y Electrónica.

DNI: Documento Nacional de Identidad

ID: identificador único

7.3.1.2 ABREVIATURAS

Req. : Requisitos

Id.: Identificador

DA: Diagrama de Actividad. Artefacto de UML para el modelado de los procesos de la organización.

DTE: Diagrama de Transición de Estados. Artefacto de UML para el modelado de la transición de estados que sufre un objeto de análisis a lo largo de los procesos del sistema.

DS: Diagrama de Secuencia. Artefacto de UML para el modelado de la interacción del Actor del Sistema con el Sistema

DC: Diagrama de Colaboración. Artefacto de UML para el modelado de la interacción de los conceptos del Sistema para dar solución a una petición del Actor del Sistema, es decir cómo se relacionan los conceptos de manera colaborativa para dar solución a una operación de sistema.

Op. : Operación del sistema.

GPL: Licencia para Software Libre de GNU (sistema operativo completamente libre)

7.3.2 REFERENCIAS

Plantilla Estándar IEEE 830 para Especificación de Requisitos de Software.

7.3.3 VISIÓN GENERAL DE LA ERS

En la Fase Inicial del Proceso de Desarrollo utilizando la metodología SCRUM se realizan de manera progresiva e iterativa la recopilación de los requisitos del nuevo sistema. Es a través de las Entrevistas, Cuestionarios y Observación de Tareas con los Usuarios del Sistema que se logra este objetivo que es fundamental para la realización del mismo.

Esta documentación resulta ser la base para la construcción del Sistema. En las distintas etapas se validara los requisitos descritos en la misma, corrigiendo los errores o implementando las mejoras.

Cabe destacar que la presente documentación se refiere a la primera etapa del desarrollo del sistema y se corresponde a las necesidades planteadas para los usuarios Clientes.

7.3.4 DESCRIPCIÓN GENERAL

7.3.4.1 PERSPECTIVA DEL SISTEMA

El sistema está basado en los conceptos de sitios web dinámicos, por lo que se dispone de los formularios y elementos de los mismos.

7.3.4.2 FUNCIONES DEL SISTEMA

- Registrar los nuevos eventos en la agenda de eventos
- Registrar los espacios físicos disponibles con sus respectivos recursos
- Mostrar los eventos públicos registrados en la pantalla de información
- Presentar la información cuando se consultan los eventos diarios, semanales, mensuales y todos los eventos.
- Presentar la información de los eventos a través de reportes para exportar en formato .xls y .pdf
- Permitir modificar los datos de los eventos
- Permitir el cambio o recuperación de usuario y/o contraseña
- Permitir modificar los datos de los espacios físicos
- Permitir cancelar los eventos registrados

7.3.4.3 CARACTERÍSTICAS DE LOS USUARIOS

TIPO DE USUARIO	CLIENTE
-----------------	---------

FORMACIÓN	no requiere
HABILIDADES	no requiere
ACTIVIDADES	Consultar la información acerca de los eventos diarios en la pantalla pública

Tabla 6 - Tipo de Usuario: Cliente

TIPO DE USUARIO	ADMINISTRADOR
FORMACIÓN	Universitario en curso o completo
HABILIDADES	Manejo de computadoras e internet
ACTIVIDADES	Registrarse en el sistema Registrar, consultar y Actualizar eventos Registrar, consultar y Actualizar espacios físicos

Tabla 7 - Tipo de Usuario: Administrador

7.3.4.4 RESTRICCIONES

Tecnológicas: la aplicación será desarrollada bajo un entorno web con lenguajes de programación con licencias libres o gratuitas.

Económicas: el proyecto no insumirá costos por la compra de licencias para los lenguajes de programación.

No habrá costos de registración ni de mantenimiento de cuentas para los usuarios Clientes.

Operativas: se dispondrá de un manual de usuario para guiar al usuario Cliente en el desarrollo de sus actividades en el sistema.

Seguridad: Por razones de seguridad, el Sistema mostrará la información adecuada a los Niveles de Usuarios previstos para determinar el tipo de información a mostrar.

El sistema brindará seguridad validando usuario y contraseña en todas las interfaces que correspondan a los niveles de Usuarios, habilitando los permisos correspondientes a cada uno. Queda bajo responsabilidad del usuario el manejo y modificación de información.

7.3.4.5 SUPOSICIONES Y DEPENDENCIAS

Suponemos que el acceso al Sistema estará siempre disponible.

La velocidad de acceso y disipación de la información de los proveedores de internet estará acorde con los estándares del mercado.

7.3.4.6 REQUISITOS FUTUROS

Solicitud de reserva de eventos online.

7.3.5 REQUISITOS DE INTERFACES EXTERNAS

7.3.5.1 INTERFACES DE USUARIO

Req.1.- —El sistema deberá proporcionar una interfaz de usuario amigable: en la que se combinen ventanas, menús y formularios propios del diseño web

Tipo: Requisito

Prioridad: Alta y Obligatoria

Descripción: La interfaz de usuario es basada en ventanas, menús y formularios para facilitar la labor del usuario, de tal modo que el uso del sistema no sea incómodo, especialmente para personas no familiarizadas con programas informáticos.

Se priorizará el uso de listas desplegables para el ingreso de datos, minimizando los errores por ingresos manuales en campos de texto.

Se procura la selección de imágenes visuales para la elección del producto con un mínimo de ingreso de datos que es en este caso el ingreso de la cantidad, esta se puede evitar seleccionando más de una vez la misma imagen.

7.3.6 INTERFACES DE HARDWARE

Req.2.- “El sistema adaptará las interfaces a los distintos dispositivos desde los cuales se accedan ”

Tipo: Requisito

Prioridad: Alta y Obligatoria

Descripción: el sistema deberá tener la propiedad de adaptar su contenido a cualquier dispositivo físico desde el cual sea accedido, como ser PC de escritorio, netbooks, Notebooks, Smartphone.

7.3.7 INTERFACES DE SOFTWARE

Req.3.- “El sistema deberá poseer la funcionalidad para imprimir a través de la impresora local que posea el dispositivo”

Tipo: Requisito

Prioridad: Alta y Obligatoria

Descripción: Esta funcionalidad es requerida para imprimir los reportes de eventos registrados en el sistema.

7.3.8 REQUISITOS DE RENDIMIENTO

Req.4.- “El sistema deberá funcionar soportando el acceso simultáneo desde distintos equipos”

Tipo: Requisito

Prioridad del Requisito: Alta, siendo obligatorio.

Descripción: El acceso simultáneo por parte de varios usuarios a la base de datos es completamente necesario.

Req.5.- “El sistema deberá responder a cualquier petición en menos de 5 segundos”

Tipo: Requisito

Prioridad del Requisito: Alta, siendo obligatoria.

Descripción: Este requisito se impone para conseguir que el sistema sea suficientemente eficiente.

Req.6.- “El sistema deberá ser capaz de manejar decenas de miles de registros”

Tipo: Requisito

Prioridad: Alta, siendo obligatorio para el funcionamiento eficiente del sistema.

Descripción: Se ha estimado que se almacenarán decenas de miles de registros en la base de datos que se encargue de mantener la información necesaria para el sistema. Por tanto, el sistema debe ser capaz de manejarlos eficientemente.

Req.7.-: “El sistema deberá tener un manual de usuario para facilitar los mantenimientos que serán realizados por el administrador”

Tipo: Requisito

Prioridad del Requisito: Alta.

Descripción: El sistema debe disponer de una documentación fácilmente actualizable que permita realizar operaciones de mantenimiento con el menor esfuerzo posible.

Req.8.- “El sistema garantizará a los usuarios un desempeño en cuanto a los datos almacenados en el sistema, ofreciéndoles una confiabilidad a esta misma”.

Tipo: Requisito

Prioridad: Alta, siendo obligatorio para el funcionamiento eficiente del sistema

Descripción: Garantizar el desempeño del sistema informático a los diferentes usuarios. En este sentido la información almacenada o registros realizados podrán ser consultados y actualizados permanentemente y simultáneamente, sin que se afecte el tiempo de respuesta.

Req.9.- “Garantiza al usuario el acceso de información de acuerdo al nivel que posee”.

Tipo: Requisito

Prioridad: Alta, siendo obligatorio para el funcionamiento eficiente del sistema

Descripción: Facilidades y controles para permitir el acceso a la información al personal autorizado a través de internet, con la intención de consultar información pertinente para cada una de ellas.

7.3.9 REQUISITOS DE DESARROLLO

Req.10.- “Para la realización del Back-End, el sistema deberá desarrollarse en el Lenguaje PHP con un Motor de Base de Datos MySQL”

Tipo: Requisito

Prioridad: Alta, siendo obligatoria.

Descripción: se prevé el Desarrollo con Lenguaje PHP, bajo la Plataforma NetBeans para PHP y con un motor de Base de Datos MySQL por ser de fácil uso y de Licencias Libres.

El motor de Base de Datos alternativo sugerido es PostgreSQL, por tratarse también de un desarrollo de software libre e incluso de código abierto.

Req.11.- “Para la realización del Front-End se implementarán el uso de tecnologías HTML5, CSS, JavaScript, JQuery.

Descripción: se prevé el Desarrollo de interfaces, utilizando el IDE de Desarrollo Sublimetext3.

7.3.10 RESTRICCIONES DE DISEÑO

7.3.10.1 LIMITACIONES HARDWARE

Req.12.-: “La configuración mínima de cada PC sobre el cual la aplicación debe funcionar es: Procesador de 200 MHz, 64 Mb RAM y 100 Mb disponibles en el Disco Rígido”

Prioridad: Alta, y obligatorio.

Descripción: esta configuración del equipo está dada por el Estudio de los requisitos de datos y los de rendimiento que manejará el sistema.

7.3.10.2 LIMITACIONES SOFTWARE

Req.13.- “La configuración de software mínima de cada PC sobre el cual la aplicación debe funcionar es: Google chrome, Internet Explorer 10.0 o Superior o Firefox 1.0 o Superior”

Prioridad: Alta, y obligatorio.

Descripción: Se obtiene del análisis de las características necesarias para ejecutar el sistema y de características de rendimiento.

Si se considerara incorporar componentes visuales se debe proveer de los Plugin necesarios para la ejecución normal de los componentes activos utilizados.

7.4 ATRIBUTOS DEL SISTEMA

7.4.1 USABILIDAD

El Sistema será dinámico y fácilmente comprensible, haciendo hincapié en las ayudas durante todo el proceso de uso que realizarán los Usuarios. Además presentará interfaces visuales amigables, en el ingreso de datos de las Interfaces Internas. Se guiará al usuario mediante mensajes de advertencia sobre los posibles errores.

7.4.2 SEGURIDAD

El Sistema será accedido por el cliente registrado mediante un Nombre de Usuario y una Clave encriptada. Cada tipo de usuario tendrá permisos según las actividades autorizadas.

Todas las operaciones del sistema serán registradas con el nombre del usuario registrado y la fecha y hora de realización.

7.4.3 FIABILIDAD

El Sistema deberá estar disponible en el horario que los comercios adheridos atiendan. Seguramente se estimará en el horario comercial, adaptándose a los horarios de bares y restaurantes.

7.4.4 DISPONIBILIDAD

Se asegurará la disponibilidad de la información en forma permanente y además se asegura la recuperabilidad a través de las copias de seguridad diarias.

7.4.5 PORTABILIDAD

El Sistema es portable para cualquier plataforma de sistema operativo: Windows, Linux, Mac, Android.

7.4.6 MANTENIBILIDAD

Al estar desarrollado como un sistema Web, las correcciones son fácilmente implementadas. Además al desarrollarse bajo Módulos es también fácilmente actualizable, pudiendo implementar nuevos Módulos para satisfacer a nuevos requisitos.

7.5 ANEXO C – PLANTILLA DE PRUEBA DE SISTEMA

Para Caso de Prueba se debe completar la siguiente Plantilla

CASO DE PRUEBA < NOMBRE DE CASO DE USO >		Fecha: __/__/____ Hora: __:____ Hoja 1 de 2
DESCRIPCIÓN: <Descripción del caso de prueba>		
HARDWARE UTILIZADO <Describir el dispositivo utilizado para la prueba>		
PASOS: <Pasos generales para la prueba, basados en los Casos de Uso propuestos en el apartado 3.2.3.>		
RESULTADO OBTENIDO: <Resultado obtenido de la ejecución del caso de prueba>		
CALIFICACION OTORGADA ... MUY SATISFACTORIA ... SATISFACTORIA ... BUENA ... REGULAR ... NO SATISFACTORIA		

CASO DE PRUEBA < NOMBRE DE CASO DE USO >		Fecha: __/__/____ Hora: __:____ Hoja 2 de 2
CUESTIONARIO DE LA PRUEBA (Obligatorio) (Marque lo que corresponda)		
SEGURIDAD Y CONTROL: Ha podido iniciar sesión normalmente? Si No Ha podido recuperar su contraseña? Si No Observaciones o Sugerencias:		
USABILIDAD E INTEGRACION Las interfaces son comprensibles? Si No Ha tenido problemas en la interpretación de las consignas? Si No Se ha podido desplazar por el sitio fácilmente? Si No Observaciones o Sugerencias:		
FUNCIONALIDAD: Pudo realizar la consigna? Si No Observaciones o Sugerencias:		
ACCESOS (CARGA Y STRESS) Cuantos accesos realizó? Puede indicarlos por franja horaria? Desde las 08 a 12 horas: _____ Desde las 12 a 18 horas: _____ Desde las 18 a 20 horas: _____ Desde las 20 a 24 horas: _____ Desde las 00 a 06 horas: _____ Desde las 06 a 08 horas: _____		

8 SOBRE EL PROYECTO

En el desarrollo del sistema web a medida, utilizaremos una metodología de desarrollo llamada SCRUM o metodología Ágil.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al cliente. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos

son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

A continuación comentamos algunos de los beneficios de utilizar esta metodología de desarrollo.

Beneficios

- Cumplimiento de expectativas: El cliente establece sus expectativas indicando el valor que le aporta cada requisito / historia del proyecto, el equipo los estima y con esta información se establece su prioridad. De manera regular, en las demos de Sprint se comprueba que efectivamente los requisitos se han cumplido y transmite se feedback al equipo.
- Flexibilidad a cambios: Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.
- Reducción del Time to Market: El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.
- Mayor calidad del software: La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- Mayor productividad: Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- Maximiza el retorno de la inversión (ROI): Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.
- Predicciones de tiempos: Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente,

es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog.

- Reducción de riesgos: El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

8.1 DETALLES DE DESARROLLO

Las tecnologías que se emplearán para el desarrollo de este proyecto son HTML5, CSS, PHP, MySQL, Javascript y Laravel 5.1.

Para la realización del proyecto, estableceremos un plan de trabajo en cinco etapas:

1. Definición de los objetivos y resultados a lograr.
2. Contenidos y arquitectura de la información
3. Maquetación y diseño web
4. Programación y puesta a punto.
5. Lanzamiento online y revisión.

1 - Definición de los objetivos y resultados a lograr

Nuestro objetivo principal es desarrollar un sistema hecho a medida que se adapte a las necesidades y requerimientos de su empresa.

Ofrecer un servicio que les ayude a mejorar la administración y control de los procesos de reserva, de manera rápida, fácil y segura.

Desarrollar un sistema escalable, el cual brinda la posibilidad de incorporar nuevas funcionalidad en el futuro, sin realizar demasiados cambios.

Desarrollar un sistema que presente pantallas comunes o similares a lo más usado en mercado actual, con utilizando interfaces amigables e intuitivas, con el objetivo de que los usuarios se sientan familiarizados de alguna manera con el sistema, de este modo se reduce el rechazo que muchas veces provoca el interactuar con algo nuevo.

2. Contenidos y arquitectura de la información

Definidos los objetivos, se procede a analizar los datos obtenidos para obtener el diseño lógico del sistema y el modelo de datos que este empleará a tales fines.

3. Maquetación y diseño web

Una vez definidos los objetivos y los resultados que se esperan lograr, así como los contenidos y la estructura de navegación web, procederemos a la maquetación y diseño de la aplicación web. El primer paso consistirá en la realización de un prototipo o wireframe con el software Balsamiq. Una vez aprobado el prototipo, se procederá a iniciar el proceso de desarrollo.

4. Programación y puesta a punto

Una vez se haya aprobado el diseño propuesto, iniciaremos el proceso de desarrollo de código. Todo el código que generamos lo hacemos de manera manual con el IDE de trabajo Sublimetext3, lo que garantiza la calidad y validez del mismo.

Nos aseguramos de que el sistema se visualice correctamente en todos los navegadores web modernos, así como en dispositivos móviles. Para ello utilizamos la técnica de mejora progresiva que optimizará la web dependiendo del dispositivo.

Aclaración: No damos soporte a Internet Explorer en versiones anteriores a la 10, por ser un navegador desfasado, ya que se creó en el año 1999. Sin embargo, en caso de querer dar soporte a este navegador, podemos hacerlo a un coste adicional, aunque no se lo recomendamos.

5. Lanzamiento online y revisión

Con la aprobación de cada Sprint, se procederá a la actualización del sistema subiendo los cambios al servidor de producción. Dispondrá entonces de 7 días para su revisión. Pasado

este plazo, se entenderá que el Sprint es correcto y por ello será finalizado siguiendo así con el curso normal del proyecto de desarrollo.

