

Administración de Proyectos Informáticos

Sitio web para eventos

Instituto Profesional AIEP – Puerto Montt

2019

Integrantes: **Juan Mansilla**

**Jaime Diaz**

**Camilo Mancilla**

**David Moralez**

****

**Tabla de contenidos**

Tabla de contenidos 3 Estudio de factibilidad 6 Planteamiento de la problemática. 6 Factibilidad. 7 Factibilidad técnica. 7 Factibilidad Operativa 11 Costo operativo 11 Riesgos 13 Ventajas 13 Desventajas 13 Factibilidad económica. 14 Costos de software y hardware 15 Costos personales 16 Costo operativo de desarrollo 17 Costo de personal 18 Costos totales 19 Resumen y Conclusiones 20 Análisis DAFO 20 Objetivos del Proyecto 21 Objetivo general 21 Objetivos específicos 22 Benchmarking 22 Estado del Arte 25 Resumen 25 Introducción 26 Eventos Locales 26 Experiencia 27 Procesos 27 Participantes 28



Valor Agregado 28 Conclusiones 28 Modelo As Is 30 Contexto de uso de la solución 31 Modelo Canvas 32 Modelo To Be 33 Usuarios y descripción de roles 35 Lista de requerimientos funcionales y no funcionales 35 Fichas resumen 36 Diagrama de secuencia 53 Usuario 53 Organizador 54 Administrador 55 Modelo Lógico 56 Confección de base de datos 56 Tabla sin normalizar 56 Primera Forma Normal (1FN) 57 Segunda Forma Normal (2FN) 57 Tercera Forma Normal (3FN) 57 Modelo Lógico 58 Modelo Físico 59 Alcances y limitaciones 60 Alcances 60 Limitaciones 60 Análisis de modelo de desarrollo y metodología a utilizar. 60 Tipos de Modelo de desarrollo 61 Modelo de Desarrollo Orientado a objetos 61 Modelo de desarrollo estructurado. 62 Modelo de desarrollo evolutivo 62 Modelo RAD o de desarrollo de aplicaciones agiles 62



Modelo de Desarrollo SCRUM 63 Modelos de Desarrollo a escoger para desarrollar software 63 Desarrollo evolutivo V/S SCRUM 64 Definición 64 Propiedades de cada uno 64 Ventajas y Desventajas 67 Ventajas de Modelo Evolutivo 68 Desventajas: 69 Tabla comparativa 70 Bibliografía 72



**Estudio de factibilidad**

Planteamiento de la problemática.

Se requiere crear un sistema de administración y organización de eventos. Este sistema está pensado para empresas a organizadoras de eventos charlas, festividades o cualquier otro evento social que pueda concurrir mucha gente. Este sistema nos permitirá administrar los eventos que ocurran en nuestra región.

El Sistema deberá permitir las siguientes funciones principales:

● El *usuario* se podrá registrar en la página con o sin redes sociales

● El *usuario* podrá buscar eventos aplicando filtros.

● El *usuario y organizador* podrá realizar reclamos.

● El usuario podrá ver la ubicación de los eventos.

● El *usuario* podrá comprar/tickets para los distintos eventos.

● El *organizador* podrá registrar y administrar sus eventos.

● El *administrador* podrá ver estadísticas de cada evento (ventas de entradas, ganancias totales entre otras).

● El *administrador* podrá administrar categorías.

Se pueden considerar las siguientes vistas del sistema: vista de administrador el cual puede administrar mediante la pagina principal el total de todo lo recaudado más un sistema de gráficos que nos permite revisar las estadísticas de compras y ventas. información de conectados a la plataforma y una sesión de gestión de la plataforma. Vista de usuario; la cual contiene acceso a la información de los distintos eventos.



La información que se entregará a los clientes sobre los distintos servicios será la siguiente.

● Fecha de inicio y fin de cada eventos.

● Ubicación con google maps de eventos.

● Tickets disponibles

● Redes sociales del evento

● Afiche promocional

Factibilidad.

Una vez descrito el planteamiento del problema es conveniente realizar un estudio de factibilidad para el proyecto de aplicación web de sistema de organización de eventos; donde se muestra el análisis técnico, operativo y económico que implican la implementación del sistema, donde se describen las necesidades que se satisfacen con la tecnología actual, justificación de los costos, beneficios, etc.

Factibilidad técnica.

Desde el punto de vista técnico para el desarrollo del proyecto son necesarios recursos tecnológicos, por ello se realizó una evaluación de la tecnología actual existente donde se recolecto información sobre esta y la posibilidad de utilizarla en el desarrollo e implantación del sistema. Para este sistema hemos considerado dos opciones que son factibles para poner en marcha nuestro sistema de Organización de eventos, a continuación daremos a conocer y comparar estas dos opciones para poder llevarlas a cabo:

**● La primera opción** para realizar nuestro sistema es a través de la alquilación de un espacio en un servidor hosting. El hosting es un servicio en línea que te permite publicar un sitio o aplicación web en Internet. Cuando te registras en un servicio de



hosting, básicamente alquilas un espacio en un servidor donde puedes almacenar todos los archivos y datos necesarios para que tu sitio web funcione correctamente.

**● La segunda opción** como grupo de trabajo es que nosotros imlementemos un Servidor web, una de las principales razones es que no tienes limite de espacio en disco para paginas web, no tienes limitaciones en cuanto a publicación de cualquier tipo de contenido, ya que todo lo haces de forma local. Tampoco es necesario subir la web cada vez que la modificas. Al igual que en el caso anterior, al ser tu propio servidor web, todo el contenido se guardaría de forma local y estaría disponible al momento de ser modificado para todos los usuarios que quieran acceder al contenido.

A continuación iremos haciendo una comparación en cuanto a recursos informaticos nevcesitaremos por cada opción:

La siguiente tabla muestra los recursos técnicos disponibles por el grupo de trabajo y los recursos que se necesitarían por cada opción:

| Hosting | | | |
| --- | --- | --- | --- |
| Recursos | Cantidad  Disponible | Cantidad  Requerida | Características |
| Estaciones de  trabajo con  acceso a  Internet. | 4 | 4 | Computadores  personales:  Notebook,  netbook, etc. |

Estaciones de trabajo en Aula de clases.

4 4 Máquinas para pruebas de

software y de la

plataforma.



| Puntos de  accesos a  internet | 4 | 4 | Por parte de  empresas móviles,  Redes AIEP, entre  otras. |
| --- | --- | --- | --- |
| Plataforma De  Hosting | 0 | 1 | Se requiere un  servicio que nos  administre hosting |

*“tabla 1: Recursos de Hardware disponibles Opción 1: Hosting”*

| Servidor web propio | | | |
| --- | --- | --- | --- |
| Recursos | Cantidad  Disponible | Cantidad  Requerida | Características |
| Estaciones de  trabajo con  acceso a  Internet. | 4 | 4 | Computadores  personales:  Notebook,  Netbook, etc. |

Estaciones de trabajo en

Aula de

clases.

Puntos de

accesos a

internet

4 4 Máquinas para pruebas de

software y de la

plataforma.

4 4 Por parte de empresas móviles,

Redes AIEP, etc.



| router | 0 | 1 | Se requiere router  para realizar  conecciones a  internet para el  servidor web. |
| --- | --- | --- | --- |
| Servidor | 0 | 1 | Se requiere  ordenador que se  utilize como  servidor. |

Cableado de Red

0 Indefinido cableado de red RJ45 para

conectar el router

con nuestro

ordenador.

*“tabla 2: Recursos de Hardware disponibles Opción 2: Servidor Propio”*

Respecto al software, el grupo de trabajo cuenta con las aplicaciones necesarias para el funcionamiento del sistema, lo cual no provoca inversión alguna. Las estaciones de trabajo operan bajo la plataforma Windows. Para el uso general del sistema en las estaciones se debe poseer las herramientas y un navegador Web:

| Hosting | | |
| --- | --- | --- |
| Cantidad | Recurso | Característica |
| Indefinido | Windows 10  Home | Sistema operativo que utilizamos en los  Notebook personales y los PC del laboratorio de trabajo. |
| Indefinido | Google Chrome,  Web Browser | Navegadores en los que probaremos la  plataforma y en la cual buscaremos  información. |



| 1 | MacOS | Sistema operativo en el cual trabaja un  integrante del grupo y trabaja el sistema. |
| --- | --- | --- |
| 4 | Licencias  gratuitas de  MySQL | Motor de la base de datos en la que cada uno de los integrantes trabajará. |
| 4 | Licencias  gratuitas de VS  Code | Editor de texto el cual se utiliza para la  realización de nuestra plataforma. |

*“tabla 3: Recursos de software Hosting.*

| Servidor Propio | | |
| --- | --- | --- |
| Cantidad | Recurso | Característica |
| Indefinido | Windows 10  Home | Sistema operativo que utilizamos en los  Notebook personales y los PC del laboratorio de trabajo. |
| Indefinido | Google Chrome,  Web Browser | Navegadores en los que probaremos la  plataforma y en la cual buscaremos  información. |
| 1  4 | MacOS  Licencias  gratuitas de  MySQL | Sistema operativo en el cual trabaja un  integrante del grupo y trabaja el sistema. Motor de la base de datos en la que cada uno de los integrantes trabajará. |
| 4 | Licencias  gratuitas de  apache | Servidor web el cual se ocupará para alojar la plataforma en un servidor. |
| 4 | Licencias  gratuitas de VS | Editor de texto el cual se utiliza para la  realización de nuestra plataforma. |



|  | Code |  |
| --- | --- | --- |

*“tabla 4: Recursos de software Servidor Próprio”*

Factibilidad Operativa

La factibilidad operativa consiste en definir si se pondrá en marcha el sistema propuesto, aprovechando los beneficios que ofrece a todos los usuarios de nuestra aplicación, Personas de la comuna que quieran participar en eventos masivos, sociales y culturales. Esta busca que el sistema tenga la capacidad para mantenerse funcionando y que sea eficiente y eficaz.

Costo operativo

En el ámbito operativo se debe realizar una evaluación de los funciones, cargos y costos por personal antes mencionado y realizar un cálculo total de lo que nos llevará mantener los recursos humanos de nuestro equipo de trabajo: A continuación se muestra una tabla con lo que debe ser considerado al implementar cada opción como también el personal que debe ser contratado para realizar el sistema, esto es una estimación de los puestos a cubrir y del salario mensual que una empresa X puede pagar (los datos de sueldos son estimaciones basadas en la media de lo que gana un profesional en Chile)

| Puesto: | Descripción Salario  Mensual | Cantidad  Personal | Total |
| --- | --- | --- | --- |
| Documentación | Encargado de  $650.000  Documentación de  proyecto en 2º plano. | 2 | $1.300.000 |



| Scrum Master jefe de proyecto. | Coordinador general en el  $1.800.000  desarrollo de Sistema. | 1 | $1.800.000 |
| --- | --- | --- | --- |
| Analista | Encargado de Brindar  $1.200.000  soporte y análisis  informáticos | 1 | $1.200.000 |
| Programador Front End  Programador Back End  Testing | Encargado de  $860.000  programación de diseño  del sistema.  Encargado de  $980.000  programación del sistema  1º plano.  Planificar y llevar a cabo  $ 740.000  pruebas de software de la  plataforma. | 2  1  2 | $1.720.000  $980.000  $1.480.000 |
| Soporte y Redes | Profesional que se  $720.000  encargue del soporte de  servidores y técnico. | 1 | $720.000 |

*“tabla 5: Costo Operativo aproximado”*

Cabe destacar que nuestra aplicación dependiendo de la opción veremos si se necesitará personal para soporte. Si aplicamos la opción de hosting, no será necesario contar con alguien que haga soporte a servidores, no así la opción 2 que si es necesario alguien que esté disponible para realizar mantención a servidores, y que vea el tema de redes.



La Aplicación web lleva a la aceptaciónn de este que de una manera sencilla y amigable cubre con las necesidades de sus usuarios, además de que proporciona informaciónón oportuna y confiable, por lo que el sistema es factible operacionalmente.

Dependiendo la Opción, utilizar el sistema tendrá sus restricciones, tales como:

● Solamente el Administrador tendrá la facultad para gestionar la plataforma. ● Los usuarios deben estar registrados en el sistema para poder utilizarlo. ● El administrador estará facultado solamente, para agregar otro tipo de categorías del sistema de eventos.

**Riesgos**

● El sistema opera a través de la internet, por lo que si esta falla el sistema no podrá funcionar.

● El administrador debe saber muy bien cómo administrar el servicio. **Ventajas**

● Mejor opción para entregar información.

● Más rápida y fácil de usar.

**●** Mayor compatibilidad al ser una aplicación web.

**Desventajas**

● Solo contiene información de eventos realizados por una empresa. ● Depende de internet, si el usuario no tiene no podrá utilizar la plataforma. ● Si el sistema corriera por servidor propio, este no cuenta con un servidor de respaldo.

Para resolver estas desventajas tenemos pensado lo siguiente:



**●** La información de los eventos puede ser más variada si expandimos al público especifico al que estará dirigido nuestro sistema.

**●** Lamentablemente se necesita internet para utilizar el sistema, pensamos que el servicio de hosting mantendrá siempre en línea el sistema para que pueda ser utilizado, si utilizáramos el servidor propio este igual debe estar siempre conectado para que el sistema corra sin interrupciones.

**●** En este caso se ve la posibilidad de considerar adquirir un servidor de respaldo para cualquier inconveniente con el sistema una vez se hayan generado recursos

Factibilidad económica.

A continuaciónn, se presenta el estudio que dio como resultado la factibilidad económica del desarrollo del nuevo sistema de información donde se determinaron los recursos económicos para desarrollar la aplicación web de sistema de organización de eventos.

Dado las dos opciones (Hosting, servidor propio), tenemos que revisar los costos de cada opción de implementación, Una vez vistos la factibilidad técnica y operativa de cada una. Al calcular la factibilidad económica de cada opción veremos cuál es la mejor para implementar.

En cuanto a gastos generales, puesto que el trabajo se realizará en aulas del instituto AIEP de Puerto Montt, es factible considerar gastos de locomoción, al igual que gastos de alimentación. Estos efectivamente correrán a cargo de cada integrante del grupo; por lo que no se consideraría parte de los gastos en común como equipo. Otro punto a tratar sería de gastos generales para imprevistos o para la impresión por trabajos o documentación; estos gastos sería factible considerarlos en nuestro plan de efectividad, Tanto para las dos opciones se requerirían los mismos costos generales:

Costos Generales: Gasto Mensual:



| Impresiones: | $1000 |
| --- | --- |
| Otros: | $1000 |

*“Tabla 6: costos Generales Hosting // Servidor Propio*

**Costos de software y hardware**

Contemplando el costo de los softwares y hardware que utilizaremos podemos considerar lo siguiente:

Las operaciones y construcción de la plataforma se realizarán en las computadoras personales de cada integrante del equipo, por lo que no se gastará en hardware. En cuanto al software utilizado serán los siguientes

| Costos de Software | | |
| --- | --- | --- |
| Descripción | Software | Costo |
| Sistema operativo | Windows 10 | Windows 10 home licencia $129.990 o imagen ISO. Costo $0 para nuestro equipo. |
| Manejador de base de datos | MySQL | Licencia gratuita. |
| Interprete de Scripts de Servidor | PHP  (Hypertext)  Pre-processor) | Licencia gratuita |

Servidor web Apache Licencia gratuita.



| Desarrollo de  Interfaces | VS Code.  HTML  CSS  JSS | Licencia gratuita ( a través de descarga en www.microsoft.com |
| --- | --- | --- |
| Servicio de Hosting | EJ:  V2Networks | $20.000 anuales Cuenta Premium |

*“Tabla 7: costos de software Hosting // Servidor Propio”*

**Costos personales**

Aquí se incluyen los gastos generados por el recurso humano que se necesita para el desarrollo del sistema únicamente. Nuestro equipo cuenta con 4 participantes para el desarrollo del sistema:

● 1 Scrum Master

● 1 Documentador.

● 1 Programador Back End.

● 1 Testeador.

De acuerdo a cada opción de implementación, si utilizáramos un servicio de Hosting entonces necesitaríamos además:

● 1 Documentador.

● 2 Programadores Front End.

● 1 Testing.

Si aplicáramos la opción de Servidor Propio, aparte de lo mencionado se necesitaría además:



● Todo lo anterior más un encargado de Redes y soporte.

Hay que analizar que para llevar a cabo todo esto necesitaríamos un lugar de trabajo fijo que podría ser un departamento o una oficina. Esto llevaría a considerar gastos de arriendo y gastos comunes (agua luz, etc.) para que el lugar en donde desarrollar sea agradable para el grupo de trabajo. Hay que considerar calefacción mayoritariamente puesto que Puerto Montt es frio la mayor parte del año. En cuanto a alimentación el equipo de trabajo se abastece cada uno propiamente con comida.

Haciendo una estimación los gastos se presentan en la siguiente tabla (los datos son estimaciones basadas en cotizaciones de gatos:

| Gasto | Descripción | Monto (Aproximado  Mensual) |
| --- | --- | --- |
| Arriendo de  Oficina | Lugar de desempeño del grupo de trabajo | $350.000 |
| Gastos Comunes | Agua, Luz, Internet, Calefacción | $180.000 |

*“Tabla 8: costos personales Hosting // Servidor Propio”*

**Costo operativo de desarrollo**

Estos costos se refieren a aquellos necesarios para la operatividad de las actividades de nuestra empresa durante el periodo en el que se realizará el proyecto. Si implementamos el servicio de Hosting, Habría que considerar el costo Mensual de una suscripción. Cotizando entre distintos servicios que ofrecen Hosting. El mejor plan tiene un valor aprox. de $20.000 anuales. Sin embargo considerando el sistema de pago a través de Paypal es necesario tener en cuenta la comisión que Paypal cobra a través de los pagos realizados por los usuarios.

| Gasto | Descripción | Monto (Aproximado  Mensual) |
| --- | --- | --- |



| Servicio de Hosting | Servicio para mantener la plataforma en linea. | $20.000 Anuales. |
| --- | --- | --- |

*“Tabla 9: costo operativo de desarrollo Hosting ”*

En cuanto a adquirir un **servidor propio** es necesario considerar que los costos operativos de desarrollo son variados. Puesto que necesitaríamos considerar adquirir el hardware necesario para que el sistema corra. Ya que este estaría instalado en un servidor propio, pero esto conlleva a gastos de luz y consumo de energía un tanto más elevados:

| Gasto | Descripción | Monto (Aproximado Mensual) |
| --- | --- | --- |
| Router | Dispositivo para conectar internet | $49.990 aprox. |
| Servidor | Ordenador o servidor para subir el sistema. | $599.990 aprox. |
| Cableado de  Red | cableado de red RJ45 para conectar el router con nuestro ordenador. | $35.990 aprox.. |

*“Tabla 10: costo operativo de desarrollo Servidor Propio ”*

**Costo de personal**

En el costo de personal se incluyen los gastos generados por recurso humano en nuestro grupo donde bajo su responsabilidad estará el contar con el personal para la operación y funcionamiento del sistema. Actualmente se requiere contratar personal extra para el desarrollo de nuestra plataforma puesto que cada integrante realiza una determinada tarea para la creación de nuestra plataforma, pero se necesitan los siguientes puestos además de:

| Puesto | Descripción: | Salario  Mensual | Cantidad de Personal | Total |
| --- | --- | --- | --- | --- |



| Documentación | Encargado de  Documentación de proyecto en 2º plano. | $650.000 | 1 | $650.000 |
| --- | --- | --- | --- | --- |
| Programador  Front End | Encargado de programación de diseño del sistema. | $860.000 | 2 | $1.720.000 |
| Testing  Soporte y Redes | Planificar y llevar a cabo pruebas de software de la plataforma.  Profesional que se encargue del soporte de servidores y técnico. | $ 740.000 $720.000 | 1  1 | $740.000  $720.000 |

*“Tabla 11: costos de personal”*

**Costos totales**

| Gasto | Descripción | Monto (Aproximado Inicial) |
| --- | --- | --- |
| Costos Generales | Imprevistos | $2000 |
| Windows 10 home | Licencia | $129.990 |
| Arriendo de Oficina | Lugar de desempeño del grupo de trabajo | $350.000 |
| Gastos Comunes | Agua, Luz, Internet, Calefacción | $180.000 |
| Servicio de Hosting | Servicio para mantener la plataforma en línea. | $20.000 Anuales. |

Puesto de trabajo faltantes

Personal que falta por completar el equipo de trabajo

$4.310.000



| **Total:** |  | **$4.991.990** |
| --- | --- | --- |

*“Tabla 12: costos totales Hosting”*

| Gasto | Descripción | Monto (Aproximado Inicial) |
| --- | --- | --- |
| Costos Generales | Imprevistos | $2000 |
| Windows 10 home | Licencia | $129.990 |
| Arriendo de Oficina | Lugar de desempeño del grupo de trabajo | $350.000 |
| Gastos Comunes | Agua, Luz, Internet, Calefacción | $180.000 |
| Servicio de Hosting  Puesto de trabajo faltantes | Servicio para mantener la plataforma en línea.  Personal que falta por completar el equipo de trabajo | $20.000 Anuales.  $5.030.000 |
| Hardware  **Total:** | Reuter, servidor, Cableado. | $685.970  **$6.397.960** |

*“Tabla 13: costos totales Hosting”*

Resumen y Conclusiones

Como pudimos observar los costos totales para la realización de nuestra aplicación, la mejor opción a considerar es la de realizar nuestro sistema mediante la implementación de **la primera opción**. Esta es la de utilizar un servicio de Hosting. Puesto que los gastos son mucho menores y no necesitaríamos a una persona como soporte técnico, esta tarea puede ser delegada a otro con un puesto no tan atareado. No gastaríamos en hardware y simplemente pagaríamos una suscripción anual que considerando la más cara sería de unos $20.000.



**Análisis DAFO**

| Amenazas Fortalezas |
| --- |
| ● Sistemas del mismo tipo y servicio.  ● Poca competencia.  ● Mercado cambiante y en constante  ● Servicio que opera en la web.  innovación.  Cualquier plataforma puede  ● Bajo mercado comercial.  ocuparlo.  Regulación del SII.  ● Calidad de servicio.  ● Fácil Implementación.  **●** Visión de objetivo claro. |
| Debilidades Oportunidades |
| ● Escases de capital.  ● Crecimiento fuerte de la industria  ● Equipo de trabajo limitado.  informática.  Tiempo de entrega corto.  ● Bajas tarifas.  Posibles inversiones. |

La demanda por un sitio web que permita organizar eventos es bastante baja, puesto que las empresas que organizan eventos masivos no son muchas y mucho menos en Puerto Montt. Hemos revisado algunas páginas de información de eventos, pero estas no cuentan con sistemas de pago de entradas y si lo hacen solo son eventos de gran escala. Nosotros crearemos un sistema que permita realizar comprar y que le permita al usuario revisar localización y venta de regalos si estos son permitidos.

Hay que tener en cuenta que la demanda es baja y como equipo tenemos debilidades, tales como poco capital y equipo de trabajo limitado. Pero esto se compensa con nuestras fortalezas y oportunidades. Por lo que estas debilidades pueden ser sanadas.



Nuestro servicio opera en la web por lo que somos capaces de llegar a varias plataformas. EL servicio es de buena calidad y Fácil de implementar, y tenemos los objetivos claros; hacia donde queremos apuntar como grupo de trabajo. Si las aplicaciones para organizar eventos no son muchas, sí que el mercado de la informática a crecido y este negocio puede ir en auge más adelante. Además, las bajas tarifas pueden ser un plus a la hora de implementar un sistema, debido a que la mayoría de software que se utilizará para implementar nuestro sistema es de licencia gratuita.

Si nuestro sistema se da a conocer más adelante podremos tener oportunidades de inversiones y que podrán llegar a llenar el vacío de capital con el que contamos actualmente.

**Objetivos del Proyecto**

Objetivo general

**●** El objetivo de nuestra aplicación Web es la de crear una plataforma en donde se puedan compartir toda la información y acceso a compra de entrada de eventos organizados en nuestra Región. En esta plataforma nuestro objetivo es que existan tres tipos de usuarios, los usuarios comunes; que son los que podrán informarse sobre eventos y actividades que se realicen en la comunidad de Puerto Montt y compra de entradas.

**●** Los Organizadores; usuarios que quieran dar a conocer eventos que ellos mismos realicen.

**●** Que sean los que Quieran organizar eventos y envíen la solicitud al administrador de la plataforma web. Y, por último, este mismo. El administrador, quien será capaz de administrar la plataforma, al cual le llegaran todas las solicitudes de los usuarios organizadores.



Objetivos específicos

**●** Dar a conocer a la comunidad en una plataforma única los distintos eventos y actividades sociales que se desarrollen en la comunidad de Puerto Montt. **●** Dar a conocer la información necesaria de los eventos registrados en nuestra plataforma para la compra de entradas y la asistencia.

**●** Fomentar a empresas organizadoras de eventos a utilizar la plataforma. **●** Crear una comunidad de usuarios para fomentar el uso de la plataforma **●** Administrar las solicitudes de eventos que se puedan producir por parte de organizadores.

**●** Crear un sistema que nos permita administrar de manera correcta la plataforma. **●** Mantener informes y análisis de la información sobre ventas y asistencia de los eventos.

**Benchmarking**

De acuerdo a lo establecido por nosotros como empresa, nuestra idea es la de crear una plataforma web que sea capaz de Entregar información sobre los eventos que se desarrollarán en nuestra región. Así como la posibilidad para organizadores de eventos de poder ingresar a nuestra plataforma los eventos que estos tienen planeado realizar. Así difundiendo una mayor variedad de eventos siempre y cuando estos cumplan con el criterio del administrador para ser difundido para su conocimiento.

Ahora bien, sabemos que este tipo de plataformas no es algo nuevo hablando globalmente. Al revisar el mercado pudimos observar que hay varias páginas de eventos en el mercado y que cumplen con este objetivo. Pero a la hora de revisar más cerca de la zona no hemos encontrado algún exponente que pueda ser utilizado como modelo a comparar, dado que aquí en la zona no es muy común encontrar páginas con esta función. Debido a esto es necesario hacer una comparativa con dos de las más grandes empresas que realizan



una función similar a la que tenemos planeado implementar, estos dos son ***PuntoTicket*** y ***Facebook.***

***PuntoTicket*** es una compañía de ventas y de distribución de entradas fundada en Santiago de Chile, con operaciones en Chile y Perú. Esta empresa fue una de las primeras compañías en vender entradas a través de Internet en Chile, comenzando a ofrecer este tipo de servicio en el año 2006 y es hoy la empresa ticketera más grande de Chile, tanto en volumen de conciertos, como en número de tickets vendidos.

PuntoTicket ha dedicado recursos significativos contra los fraudes: sus boletos tienen código de barras únicos, de modo que los boletos falsificados sean rechazados en la puerta de los eventos. Estos venden entradas de distintos tipos de eventos: conciertos, obras de teatro, eventos profesionales, deportes y más. El principal canal de venta de la empresa es el canal virtual, tanto web como móvil, canales con los que vende más de dos millones de entradas cada año.

Con los años, PuntoTicket ha logrado negociar acuerdos exclusivos con lugares donde se realizan eventos, por lo cual hoy es la ticketera oficial de Movistar Arena, Teatro Oriente, Teatro Cariola y más. Además, vende en exclusiva entradas de la Selección de fútbol de Chile, de los partidos de la Universidad de Chile y de los principales festivales de música que se realizan en Chile, como Lollapalooza Chile, Festival de Viña, Festival Primavera Fauna y Frontera Festival.

Sabemos que PuntoTicket es un grande de la industria en cuanto a venta de entradas en nuestro país, pero a la hora de comparar entre empresas que se dediquen a la venta de entradas de eventos, es imposible no referirse a ellos. PuntoTIcket como su descripción cita; es la empresa ticketera más grande de nuestro país. Sin embargo, esta solamente abarca la región metropolitana, por lo que si alcance alrededor del país no es grande. En cambio, nuestra idea principal sería la de poder, APARTE DE VENDER entradas; dar la



oportunidad de usuarios organizadores dar a conocer sus eventos y poder vender entradas igualmente.

Nuestra idea sería implementar in sistema similar al de PuntoTicket para vender las entradas del evento. Implementando un sistema de pago de PayPal y de tarjeta de crédito bancaria. En donde nuestros usuarios puedan comprar entradas y a la vez dependiendo del tipo de entrada comprada incorporarle beneficios y bonos. (Ej.: Un evento de anime puede tener dos tipos de entrada, entrada normal y entrada VIP; al momento que el usuario compre entrada VIP este puede optar a regalos o accesorios extras.).

Punto Ticket cuenta con un sistema de generación de código de barras únicos, el cual funciona en contra de fraudes y engaños. Podemos crear un sistema similar que le creará una entrada electrónica al usuario con un sistema de código de barras y a la vez que esta sea enviada al correo electrónico del usuario. Cabe mencionar que nosotros trabajaremos con los usuarios de tipo Organizadores que nos enviarán sus solicitudes para subir su evento a nuestra plataforma.

A la hora del sistema de ventas de punto ticket, estos cuentan con tiendas físicas las cuales venden las entradas; poseen convenios con otras tiendas para la compra de entradas y envían por correo las entradas a usuarios de otras regiones. En este sentido; debido a que la plataforma está centrada en funcionar de manera local (en la región específicamente), nuestro sistema de ventas será solamente a través de la plataforma web. Al momento de comprar las entradas estas serán generadas vía plataforma y enviadas al correo electrónico del usuario. Hay que distinguir que en ese sentido poseemos un plus en cuanto a punto ticket, debido a que ahorramos en gastos de mantención de una empresa.

Como podemos observar en la comparativa hacia la empresa PuntoTicket, Ofrecemos a los Organizadores la oportunidad de ingresar sus eventos nuestra plataforma y vender las entradas (dependiendo si para asistir al evento se necesita comprar entrada) y a la vez a los usuarios les entregamos toda la información necesaria del evento. Añadiendo



un extra de que nuestra plataforma está dirigida más al pequeño mercado de cualquier tipo de evento, no solo conciertos a gran escala en la región metropolitana.

Pasando al otro punto de referencia que tomamos para realizar el Benchmarking, está Facebook. La red social más grande a nivel global cuenta con una funcionalidad que permite a los usuarios de Facebook saber sobre eventos que se realizarán. El usuario de Facebook podrá Crear un evento directamente desde su página y, luego, difundirlo con los anuncios de respuestas a eventos de Facebook para hacer que su evento sea visible. Pueden elegir el público que podría asistir a su evento en función de ciertos datos, como la edad, el sexo, los intereses y el lugar donde viven. Esta lista de eventos que se muestran en el lado superior derecho estará configurada de acuerdo a las variables de localidad, amigos y amigos de amigos que asistirán al evento. También en base a preferencias que posee el usuario de Facebook, la red social le sugerirá al usuario los eventos que el algoritmo encuentre que puedan interesarle. En base a esta referencia nuestra plataforma va un paso más allá y a los usuarios registrados el sistema les dará a conocer todos los eventos que estarán disponibles en nuestra plataforma.

Si bien el sistema de Facebook ofrece estas funciones, no es completo, ya que no puedes comprar entradas y no aporta mucha información referencial sobre los eventos. Nuestra plataforma puede aportar más debido a que está basada en el negocio de compra de entradas y promoción de eventos, y Facebook no se enfoca tanto en estas variables, puesto que su mayor enfoque es prácticamente compartir con amigos de la red social.

**Estado del Arte**

Resumen

Conocer el estado actual del mercado potencial en la Región de Los Lagos es fundamental a la hora de lanzar nuestra aplicación Web, que antes mencionada en el Benchmarking está enfocada a tres tipos de usuario (Administrador, Organizadores y Usuarios comunes). Nuestra competencia directa es a grandes rasgos las redes sociales



(Facebook y Twitter específicamente) y cualquier empresa dedicada al desarrollo de plataformas organizadoras de eventos. En cuanto se trata de esto estamos seguros que nuestra plataforma abarca muchas funcionalidades que comparados a otras plataformas vistas no cuentan con todo lo que nosotros ofrecemos, ya sea; Opción de subir eventos, compra de entradas, localización geográfica de eventos, Información al día de eventos realizados, biografía de expositores, etc.

Introducción

El estudio del benchmarking arrojó una comparativa a nivel regional en cuanto a nuestro mercado competitivo en el que se dio a conocer que; en cuanto a especificaciones nuestra plataforma ofrecería muchas más funcionalidades que otros competidores, y en cuanto a competencia más reconocible tenemos a Facebook que cuenta con especificaciones mencionadas en el Benchmarking.

De acuerdo a lo mencionado debemos realizar un estudio del mercado en general y el potencial valor agregado que nuestra plataforma web puede agregar al mercado actual. Sin tener en cuenta lo que puede ser la competencia menos visible hay dos empresas con las cuales podemos llegar a compararnos (Facebook y PuntoTicket) pero como lo mencionamos antes; en el caso de Punto Ticket esta ópera en Santiago. Facebook, por el contrario, no posee un sistema de pago para compra de entradas, por lo que su función es prácticamente informativa.

Eventos Locales

La premisa es simple, entregar al mercado potencial una plataforma en la cual ellos puedan acceder a toda la información disponible de la zona sobre eventos a realizarse. Esto engloba una variedad de eventos aparte de los más recurrentes (Conferencias, cursos, charlas, conciertos, ferias, etc.) Estos estarán separados por categorías a los cuales los usuarios registrados en la plataforma podrán acceder y poder obtener toda la información necesaria además de poder comprar entradas vía online, en donde nuestra plataforma



generará un a entrada con un código de barras, que será enviada al correo de los usuarios. Además, contaremos con usuarios organizadores que podrán subir sus eventos a la plataforma web.

La premisa es simple, como dijimos anteriormente, pero puede llegar a convertirse en una gran oportunidad de negocios. En la zona no contamos con una página que pueda llegar a abarcar todas las funcionalidades que nosotros Queremos desarrollar. Y además la zona no cuenta con un conglomerado de eventos demasiado grande y vistoso como los que se realizan en la Región Metropolitana, pero pensamos que esto puede ayudar a mejorar esa situación en nuestra región.

Experiencia

Como grupo pudimos conversar con un organizador de eventos. Al hablar con él y plantearle nuestra visión; esta persona nos comentó del sistema que la emplea a la hora de darse a conocer como organizador. En nuestra entrevista, esta persona nos comentaba el hecho de que poseía una página creada en su perfil de Facebook, en la cual el subía anuncios y fotos de eventos que él había organizado, etiquetando amigos y pidiendo de que etiqueten amigos de sus amigos. Hoy claramente esto puede considerarse como Spam. Los organizadores a su vez contratan servicios para poder crear una página particular y servicios de hosting para poder mantenerse en línea. Al comentarle sobre nuestras funciones que planeábamos implementar en nuestra plataforma, la visión de esta persona fue muy importante para poder desarrollar nuestras ideas. Puesto que le comentamos lo que queríamos diseñar, el organizador encontró favorable la idea de poder subir sus eventos a la página y que esta pueda enfocarse principalmente a la información de eventos a realizarse. Le explicamos que él podría contar con un perfil de organizador y que podrá subir sus eventos a la página, no sin antes ser revisado por el administrador del sitio, para verificar que este no incumpla con las normas. “Es una política bien definida en cuanto a seguridad” nos comentó. “Puesto que siempre hay un graciosito que le gusta trolear por internet”.



Procesos

Cabe mencionar los sitios de eventos específicos (Véase: FestiGame, ComicCon, entre otros). Estos Poseen paginas propias y el proceso para comprar entradas es similar al de puntoTicket:

● Entras a la página.

● Te informas del evento y si tiene que pagar entradas.

● Si es así, compras tu entrada.

● Te envían las entradas por correos de chile.

Un proceso simple, con la diferencia que nosotros queremos que la experiencia de usuario sea más fácil y a la vez más seguro; debido a esto implementamos un sistema en el cual el usuario podrá pagar y las entradas se enviaran por correo electrónico. Además de contar con arios sistemas de pago.

Participantes

En esta plataforma nuestro objetivo es que existan tres tipos de usuarios, los usuarios comunes; que son los que podrán informarse sobre eventos sociales, charlas y actividades que se realicen en la comunidad de Puerto Montt y compra de entradas. Los Organizadores; usuarios que quieran dar a conocer eventos que ellos mismos realicen. Que sean los que Quieran organizar eventos y envíen la solicitud al administrador de la plataforma web. Y por último, este mismo. El administrador, quien será capaz de administrar la plataforma, al cual le llegaran todas las solicitudes de los usuarios organizadores.

Valor Agregado

En cuanto a las ganancias de nuestra plataforma se vería de varias maneras. Vender la plataforma a alguna empresa dedicada a esto es una opción. Cobrar una pequeña



comisión por cada venta de entrada es la idea principal, pero hay que tener en cuenta los costos de implementación vistos en el estudio de factibilidad, por lo que al realizar este estudio aún no se tiene muy claro este punto.

Pero el punto más importante a recalcar es que si nuestra plataforma se lanza al mercado, esta puede llegar a beneficiar de maneras positivas tanto a los organizadores como a los usuarios. Debido a que estos tendrán una plataforma principal para que los eventos se den a conocer en la zona y se pueda potenciar la venta de entradas para eventos pagados, y a la vez potenciar la accesibilidad de la información de eventos a realizarse en nuestra región.

Cabe destacar que siempre hay que empezar por lo más sencillo por lo que la plataforma estaría ligada a Puerto Montt en un Inicio. A medida que esta se valla dando a conocer la posibilidad de abarcar más ciudades es factible.

Conclusiones

El Benchmarking realizado nos da la idea para poder implementar nuestra plataforma. Con un mercado completivo inexistente en la región; donde la competencia directa serían el gigante global Facebook, o la empresa chilena PuntoTicket pero a un nivel aún bajo el promedio (puesto que no cumplen con las mismas funciones que nosotros) es factible considerar que no tenemos muchas dificultades a la hora de entrar al mercado. Pero si hay que tener los ojos puestos en el mercado creciente. Por muy buena que sea nuestra idea siempre ay que tener en cuenta que puede aparecer alguien con nuestra misma idea y puede llegar a implementarla mejor. En resumen, tenemos una idea buena pero quizás a lo largo del desarrollo de esta sería bueno siempre ir haciendo estudios del mercado existente e ir innovando en nuestra plataforma. Para así poder seguir manejando bien el mercado de la zona y poder generar buenas ganancias a largo del tiempo.



**Modelo As Is**

|  |
| --- |

*Ilustración 1: Modelo As Is*

**

**Contexto de uso de la solución**

El proyecto busca llevar información sobre eventos a las personas, teniendo también la oportunidad de administrar eventos. Todo esto se puede realizar de manera eficiente a través de la web, ya que hoy en día es una plataforma a la que casi la totalidad de las personas tenemos acceso. Consideramos importante que la gente pueda asistir a ciertas actividades de su interés, poder mantenerse informado sobre eventos que se realicen cerca, muchas veces no estamos al tanto de esto.

Uno de los puntos más importantes a tomar en cuenta es la interacción del usuario con el sistema, que puedan ingresar con datos de usuario para poder administrar de forma personalizada la asistencia de eventos, o si desea, crear un evento.

El presente modelo señala el funcionamiento del sistema como un participante llamado “internet” aparte al usuario y organizador. El problema se encuentra en que en este Pool no sabemos que es lo que está pasando realmente, cuando un usuario quiere comprar un ticket, este debe ser registrado, procesar la solicitud y estar preparado para posibles errores.

La idea es describir el proceso más detalladamente, comentando también de que trata cada tarea para evitar mal entendido de parte del personal, por ejemplo, a la hora de comprar un ticket primero debe ser elegido por el usuario, y para esto el sistema cuenta con categorías para hacer más fácil la navegación.

Otro proceso que le hace falta descripción es a la hora de administrar eventos, de esta tarea se encargan los organizadores. Lo importante de este proceso es que los eventos son registrados por los organizadores, es prácticamente la base del proyecto.



Mientras un sistema funcione a la perfección no nos podríamos enterar de que existen errores, es solo hasta que ocurren cuando nos enteramos de que hay que buscar una solución. El modelo no considera ciertas situaciones que tienen altas probabilidades de que ocurran, por ejemplo, a la hora de comprar un ticket si el usuario no está registrado, no es factible enlistarlo en un evento de forma anónima, en este caso la idea es que el usuario se registre y deje sus datos en la ficha de usuarios que van a asistir al evento.

Una vez solucionadas estas carencias de información, el proceso puede fluir de mejor manera y también tener un plan en caso de que existan errores. El modelo to be integraría muchas más información y solución de ciertos errores como el enlistado de usuarios anónimos, detalle del proceso de compra del usuario y descripción de ciertas tareas.

**Modelo Canvas**

| Socios claves | Actividades  claves | Propuesta de valor | Relación  con cliente | Segmento de  clientes |
| --- | --- | --- | --- | --- |
| -  Organizaciones , empresas y personas que realicen  eventos dentro de la región. | - Desarrollo de la plataforma - Implementac ión de  métodos de  pago  - Generar base de datos  - Testeo del  producto | Promoción de  eventos online:  Para usuarios:  - Facilidad de uso - Facilidades de  pago  - Control y registro de cada compra  Para organizadores: | - Atención  personalizada ente  cualquier  problema  (tanto para  usuarios  como  organizadore s). | - *Usuarios* interesa das en asistir a  eventos dentro de la región.  - *Organizaciones, empresas y*  *personas* que  realicen eventos dentro de  la región |



|  |
| --- |
| Recursos  claves |

- Profesionales en TIC y

desarrollo

- Herramientas de desarrollo de software

- Hosting web

- Poder

de auto administrar c ada evento.

- Registro de

ganancias totales.

| - Lista de  preguntas  frecuentes |
| --- |
| Canales |

- Sitio web oficial de la pagina

Estructura de coste Fuentes de Ingreso

| - Costo desarrollo de software  - Ganancias obtenidas gracias a través de  - Personal  porcentaje por venta de entradas  - Costo por hosting web  - Ganancias por publicidad anexa a la  pagina |
| --- |

**Modelo To Be**

****

|  |
| --- |

*Ilustración 2: Modelo To Be*

|  |
| --- |

*Ilustración 3: Subproceso compra de entradas*

**

|  |
| --- |

*Ilustración 4: Subproceso administrar eventos*

**Usuarios y descripción de roles**

| Usuario | Roles |
| --- | --- |
| Administrado r | ● Encargado de administrar cada reclamo o sugerencia realizado por los usuarios.  ● Encargado de administrar a todos los usuarios y categorías registradas  ● Responsable de aprobar o desaprobar cada solicitud de organizador |
| Organizador | ● Sera capaz de registrar y administrar eventos de su autoría. |
| Usuario | ● Buscar eventos, comprar tickets y administrar cada transacción realizada con su cuenta. |



**Lista de requerimientos funcionales y no funcionales**

| Requerimientos Funcionales | ID |
| --- | --- |
| El usuario podrá registrarse con sus datos personales | RF1 |
| El usuario podrá buscar eventos aplicando filtros | RF2 |
| El usuario podrá comprar tickets/entradas a través de la plataforma | RF3 |
| El usuario podrá administrar los tickets/entradas comprados | RF4 |
| El usuario podrá realizar reclamos y sugerencias | RF5 |
| El usuario podrá ver el estado de los reclamos | RF6 |
| El organizador podrá solicitar una cuenta | RF7 |
| El organizador podrá registrar sus eventos | RF8 |
| El organizador podrá administrar sus eventos  El organizador podrá realizar reclamos y sugerencias | RF9  RF10 |
| El organizador podrá generar informes sobre eventos registrados El administrador podrá aprobar y desaprobar organizadores | RF11  RF12 |
| El administrador podrá administrar las cuentas de los usuarios y organizadores El administrador podrá abrir y cerrar reclamos o sugerencias | RF13  RF14 |
| El administrador podrá administrar categoría  El administrador podrá registrar un nuevo administrador | RF16  RF17 |
| Los usuarios podrán iniciar y cerrar sesión | RF18 |
| Los usuarios podrán modificar sus datos personales | RF19 |
| Los usuarios podrán recuperar sus contraseñas | RF20 |
| Requerimientos Funcionales | ID |
| El sistema debe proporcionar mensajes de error que sean informativos, claros y bien formados. | RNF  1 |
| El sistema debe contar con un módulo de Preguntas frecuentes. | RNF  2 |



| La aplicación web debe poseer un diseño “Responsivo” a fin de garantizar la adecuada visualización en múltiples dispositivos | RNF  3 |
| --- | --- |
| El sistema debe poseer interfaces gráficas bien formadas. | RNF  4 |
| El sistema debe contar con manuales de usuario | RNF  5 |

Fichas resumen

| Nombre  Escenario | Registro del usuario y organizadores |
| --- | --- |
| Requerimientos | RF1, RF7 |
| Actores | Usuario, organizador |
| Precondiciones | ● El **usuario** y el **organizador** deberán tener una red social o un correo electrónico que no se encuentre registrado en el sistema. |
| Descripción  Escenario | ● El **usuario** para poder adquirir o administrar sus tickets para cualquier evento deberá registrarse con los siguientes datos:  ● Correo electrónico  ● Nombre y apellido  ● Teléfono  ● Contraseña  También el usuario tendrá la posibilidad de registrar con las siguientes redes sociales:  ● Facebook  ● Google |



|  | ● El **organizador** se registra en el sistema a la espera de la aprobación de un administrador, una vez aprobada el organizador podrá registrar y administrar sus eventos, los datos requeridos para la obtención de una cuenta de  organizador son:  ● Correo electrónico  ● Contraseña  ● Teléfono  ● Redes Sociales  También el organizador tendrá la posibilidad de registrar con las siguientes redes sociales:  ● Facebook  ● Google |
| --- | --- |
| Postcondiciones | ● El **usuario** podrá ingresar con su cuenta en el sistema para adquirir y administrar boletos para eventos.  ● El **organizador** podrá ingresar con su cuenta en el sistema y registrar y administrar sus eventos. |
| UML |  |



| Nombre  Escenario | Búsqueda de eventos |
| --- | --- |
| Requerimientos | RF2 |
| Actores | Usuario |
| Precondiciones | ● El usuario debe seleccionar los filtros que estime pertinente dependiendo de lo que requiera. |
| Descripción  Escenario | ● Un **usuario** registrado o no registrado al momento de buscar un evento podrá aplicar los siguientes filtros:  ● *Fecha*  ● *Categorías*  ● *Deportivos: Torneos, competencias, cursos*  *relacionados al deporte, etc.*  ● Académicos: *Ferias, congresos,*  *convenciones, exposiciones, seminarios, etc.*  ● Corporativos: *Lanzamientos,*  *inauguraciones, concursos, etc.*  ● Espectáculos: *Conciertos, festivales,*  *desfiles, espectáculos artísticos, etc.*  ● Religiosos: *Retiros, cursos religiosos,*  *campamentos, estadías, etc.*  ● Políticos: *Debates, congresos, lanzamientos*  *de campaña, etc.*  ● *Ciudad (Dentro de la región)* |
| Postcondiciones | ● El sistema mostrara al usuario todas las opciones de acuerdo a los parámetros escogidos. |



|  |  |
| --- | --- |

| Nombre  Escenario | Compra de tickets |
| --- | --- |
| Requerimientos | RF3 |
| Actores | Usuario |
| Precondiciones | ● El usuario debe buscar un evento de su interés, elegir un tipo de entrada/ticket y un medio de pago. |
| Descripción  Escenario  Postcondiciones | ● El **usuario** registrado una vez seleccione un evento y tipo de entrada/ticket deberá elegir uno de los siguientes medios de pago para continuar con la compra del ticket/entrada. ● *PayPal*.  *● Tarjeta de crédito*  *● Transferencia*  EL sistema enviará automáticamente una boleta electrónica al usuario que realiza la compra que servirá como  ticket/entrada.  ● El sistema registra la venta y notifica al usuario que esta se realizó exitosamente. |



|  |  |
| --- | --- |

| Nombre  Escenario | Administrar tickets comprados |
| --- | --- |
| Requerimientos | RF4 |
| Actores | Usuario |
| Precondiciones | ● El usuario debe ingresar con sus credenciales en la plataforma y tener una compra registrada. |
| Descripción  Escenario | ● El **usuario** al ingresar con su cuenta podrá tener acceso a una lista que registrara cada compra, de cada compra se mostraran los siguientes datos:  ● Nombre del evento  ● Fecha y hora del evento  ● Numero ticket (ID)  ● Tipo de ticket/entrada  ● Precio del ticket  Además de la opción de cancelar su ticket (Si está dentro de los plazos de cancelación establecidos por el organizador del evento) y descargar la boleta electrónica (la cual es  utilizada como ticket de entrada). |
| Postcondiciones | ● El usuario al cancelar el ticket reembolsara si se encuentra dentro de los plazos definidos y notificara el resultado. |



|  |  |
| --- | --- |

| Nombre  Escenario | Realizar reclamos y sugerencias |
| --- | --- |
| Requerimientos | RF5, RF10 |
| Actores | Usuario, organizador |
| Precondiciones | ● El **usuario** y el **organizador** deben ingresar con sus credenciales además tener un problema o sugerencia. |
| Descripción  Escenario | ● El **usuario** al ingresar con su cuenta tendrá la opción de abrir un ticket de reclamo/sugerencia seleccionando entre las siguientes opciones:  ● Reportar un problema con el pago  ● Reportar un problema con el evento  ● Reportar un error en la plataforma  ● Reportar otro tipo de error  ● Realizar una sugerencia  ● Realizar una consulta  En el *primer* y *segundo* caso se cargará una lista con tickets o eventos vigentes de los cuales se registre una compra para que puedan ser anexados como información adicional. |



|  | ● El **organizador** al ingresar con su cuenta tendrá la opción abrir un ticket de reclamo/sugerencia seleccionando entre las siguientes opciones:  ● Reportar un problema al registro de un evento  ● Reportar un error en la plataforma  ● Reportar otro tipo de error  ● Realizar una sugerencia  ● Realizar una consulta |
| --- | --- |
| Postcondiciones | ● El sistema registra el ticket de sugerencia/reclamo y queda a la espera de que un administrado responda. |

| Nombre  Escenario | Ver el estado de los reclamos |
| --- | --- |
| Requerimientos | RF6 |
| Actores | Usuario, Organizador |
| Precondiciones | ● El **usuario** o **organizador** deberá haber abierto un ticket de reclamo/sugerencia. |



| Descripción  Escenario | ● El **usuario** y el **organizador** al ingresar con su cuenta podrá acceder a una sección donde podrá responder y ver el estado de sus tickets de reclamos/sugerencias, los estados pueden ser los siguientes:  ● *En espera*: Aun no se procesa por ningún  administrador, por lo que no se puede responder.  ● *Abierto*: Procesada y contestada por un administrador, en este estado se puede responder solo hasta que el  ticket cambie su estado a cerrado.  ● *Cerrado*: En este estado se podrá responder. |
| --- | --- |
| Postcondiciones | ● El sistema permitirá mostrar el estado actual de cada uno de los tickets. |
|  |  |

| Nombre  Escenario | Registrar sus eventos |
| --- | --- |
| Requerimientos | RF8 |

Actores Organizador



| Precondiciones | ● El **organizador** debe ingresar con sus credenciales en la plataforma e ingresar todos los datos requeridos para el registro del evento. |
| --- | --- |
| Descripción  Escenario | ● El **organizador** podrá registrar un evento para el cual se necesitarán los siguientes datos:  ● Nombre del evento  ● Detalle del evento  ● Categoría del evento  ● Fecha del evento (Inicio y fin)  ● Ubicación del evento  ● Días hábiles dispuestos para cancelar un evento (del 1 al 5)  ● Redes sociales promocionales  Además de cada evento se pueden registrar diferentes tipos de tickets/entradas, de cada entrada se debe detallar:  ● Tipo de entrada (Nombre)  ● Detalle por tipo de entrada (En que se diferencia una de otra perteneciente al mismo evento)  ● Valor por entrada  ● Cantidad de entradas por tipo (Disponibilidad) |
| Postcondiciones | ● El sistema registra el evento y notifica al organizador que se registró. |





| Nombre  Escenario | Administrar eventos |
| --- | --- |
| Requerimientos | RF9 |
| Actores | Organizador |
| Precondiciones | ● El **organizador** debe ingresar con sus credenciales en la plataforma y registrar al menos un evento. |
| Descripción  Escenario | ● El **organizador** podrá administrar todos los datos  pertenecientes a los eventos vigentes que el organiza, los datos que podrá modificar son los siguientes:  ● Nombre evento  ● Detalle evento  ● Fecha del evento  ● Categoría de evento  ● Ubicación del evento  Además de cada evento se podrá modificar los datos de los diferentes tipos de tickets y entradas, los datos que podrán ser  modificados serán: |



|  | ● Detalle por tipo de entrada  ● Valor por entrada  ● Cantidad de entradas por tipo (Disponibilidad)  Cada modificación será notificada al cliente y se le otorgará un día hábil más (En el caso de que ya no disponga de días hábiles) para ver si acepta los nuevos cambios. |
| --- | --- |
| Postcondicione s | ● Cualquier modificación en un evento será registrada en el sistema notificando al organizador. |
|  |  |

| Nombre  Escenario | Generar informes de eventos |
| --- | --- |
| Requerimientos | RF11 |
| Actores | Organizador |
| Precondiciones | ● El organizador debe ingresar con sus credenciales en la plataforma y tener al menos un evento registrado. |
| Descripción  Escenario | ● El **organizador** podrá descargar informes (pdf, word, etc) de los siguientes datos perteneciente a un evento:  ● Informe con todos los detalles de un evento |



|  | ● Informe con la lista de usuario perteneciente a un evento |
| --- | --- |
| Postcondiciones | ● El sistema generara un informe en el formato seleccionado. |
|  |  |

| Nombre  Escenario | Aprobar y desaprobar organizadores |
| --- | --- |
| Requerimientos | RF12 |
| Actores | Administrador |
| Precondiciones | ● El **administrador** debe ingresar con sus credenciales en la plataforma y tener solicitudes de organizadores pendientes. |
| Descripción  Escenario | ● El **administrador** al ingresar al sistema dispondrá de una lista de solicitudes de organizadores, el administrador se encargará de aprobar o desaprobará cada solicitud verificando la información enviada. (teléfono, redes sociales, correo e información adicional agregada a la hora del registro) |
| Postcondiciones | ● El sistema notificara a través de un correo electrónico al organizador que se aprobó o desaprobó la solicitud y se activara su cuenta. |





| Nombre  Escenario | Administrar a los usuarios y organizadores |
| --- | --- |
| Requerimientos | RF13 |
| Actores | Administrador |
| Precondiciones | ● El **administrador** debe ingresar con sus credenciales en la plataforma además deben existir usuarios u organizadores registrados. |
| Descripción  Escenario  Postcondiciones | ● El **administrador** al ingresar al sistema dispondrá de una lista con todos los usuarios y organizadores que se  encuentren registrados en la plataforma, aquí podrá dar de baja tanto a usuarios como organizadores.  ● El sistema dará de baja al usuario u organizador,  notificándoles la próxima vez que inicien sesión. |



|  |  |
| --- | --- |

| Nombre  Escenario | Abrir y cerrar reclamos o sugerencias |
| --- | --- |
| Requerimientos | RF14 |
| Actores | Administrador |
| Precondiciones | ● El **administrador** debe ingresar con sus credenciales en la plataforma además deben existir al menos un ticket de reclamo/sugerencia. |
| Descripción  Escenario | ● El **administrador** al ingresar al sistema podrá administrar (Cerrar, abrir, responder) cada ticket de reclamo/sugerencia realizados por usuarios como por organizadores. |
| Postcondiciones | ● El sistema notificara y actualizara cualquier cambio de estado ocurrido al usuario que abrió el ticket. |
|  |  |



| Nombre  Escenario | Administrar categorías |
| --- | --- |
| Requerimientos | R16 |
| Actores | Administrador |
| Precondiciones | ● El **administrador** debe ingresar con sus credenciales en la plataforma. |
| Descripción  Escenario  Postcondiciones | ● El **administrador** al ingresar al sistema podrá registrar nuevas categorías, para cada categoría se solicitará:  ● Nombre de la categoría  Si se registra una nueva categoría esto se notificará a todos los organizadores.  Además podrá modificar el nombre a todas las categorías existentes. ● El sistema registrara la nueva categoría notificando al administrador. |

| Nombre  Escenario | Agregar administradores |
| --- | --- |
| Requerimientos | R17 |

Actores Administrador



| Precondiciones | ● El **administrador** debe ingresar con sus credenciales en la plataforma. |
| --- | --- |
| Descripción  Escenario | ● El **administrador** al ingresar al sistema podrá registrar otros administradores, los datos solicitados para el registro son:  ● Nombre y apellido  ● Correo electrónico  ● Contraseña |

Postcondiciones ● Se registrará el nuevo administrador en el sistema y se le enviara un correo al nuevo administrador.



| Nombre  Escenario | Iniciar y cierre de sesión |
| --- | --- |
| Requerimientos | R18 |
| Actores | Administrador, usuario, organizador |
| Precondiciones | ● El usuario debe tener una cuenta registrada o activa en la plataforma. |
| Descripción  Escenario | ● El **administrador**, **usuario** y **organizador** podrán iniciar sesión y acceder a las funcionalidades propias de cada tipo de usuario, si el registro fue realizado sin redes sociales, deberá |



|  | ingresar con su correo electrónico y su contraseña, si fue con una red social deberá elegir la red social con la cual se registró para poder ingresar a la plataforma.  Cuando estime pertinente se podrá cerrar sesión. |
| --- | --- |
| Postcondiciones | ● Al iniciar sesión el sistema autentifica y permite acceder a las funcionalidades propias de cada tipo de usuario,  ● Antes de cerrar sesión sistema preguntara al usuario si está seguro y procede con el cierre. |
|  |  |

| Nombre  Escenario | Modificar Datos |
| --- | --- |
| Requerimientos | R19 |
| Actores | Administrador, usuario, organizador |
| Precondiciones | ● Los diferentes tipos de usuario deben tener una cuenta. |
| Descripción  Escenario | ● Los **usuarios**, **organizadores** y **administradores** pueden modificar sus datos de registro, los datos que podrán ser modificados son los siguientes:  ● Nombre y apellido (Usuario, organizador y  administrador) |



|  | ● Correo electrónico (Usuario, organizador y  administrador)  ● Contraseña (Usuario, organizador y administrador) ● Teléfono (Usuario, organizador)  ● Redes Sociales (En el caso de los organizadores)  Si el usuario o el organizador se registra con redes sociales esta podrá ser desvinculada y además no podrá modificar las opciones anteriormente mencionadas exceptuando redes sociales y teléfono. |
| --- | --- |
| Postcondiciones | ● El sistema registrara y notificara al usuario cuando se realiza el cambio. |

| Nombre  Escenario | Recuperar contraseña |
| --- | --- |
| Requerimientos | R20 |
| Actores | Usuario, organizador |
| Precondiciones | ● El usuario debe tener una cuenta registrada o activa en la plataforma. |
| Descripción  Escenario | ● Los **usuarios** y **organizadores** registrados sin redes sociales si no recuerdan su contraseña podrán recuperarla ingresando su correo electrónico. |



| Postcondiciones | ● El sistema el enviara a su correo una nueva contraseña |
| --- | --- |
|  |  |

**Diagramas de secuencia**

Usuario



*Ilustración 5. Diagrama de secuencias correspondiente al Usuario*

**

Organizador

*Ilustración 6:Ilustración 5. Diagrama de secuencias correspondiente al Organizador*

**

Administrador



*Ilustración 7: Ilustración 5. Diagrama de secuencias correspondiente al Administrador*

**

**Modelo Lógico**

*****Ilustración 8: MER - Modelo Lógico*

**Confección de base de datos**

Tabla sin normalizar

| Usuario | Nombre, Apellido, Correo, Contraseña, FechaRegistro, Estado, Tipo, Redes\_sociales, , idRed\_Usuario |
| --- | --- |
| Ticket | Titulo, Detalle, Valor, Disponibilidad, Cantidad, Fecha |
| Evento | Detalle, Fecha\_inicio, Fecha\_fin, Ubicacion, Disponibilidad, Promocional, Categoria, Redes\_sociales, url |
| Reclamo | Titulo, Detalle, Fecha, Tipo\_Reclamo, Estado |
| Respuesta | Respuesta, Fecha |



| Detalle\_venta | Total, Fecha |
| --- | --- |
| Ticket\_venta | Codigo\_barra, Tipo\_Pago |

Primera Forma Normal (1FN)

| Usuario | idUsuario, Nombre, Apellido, Correo, Contraseña, FechaRegistro, Estado, Tipo, Redes\_sociales, |
| --- | --- |
| Ticket | idTicket, Titulo, Detalle, Valor, Disponibilidad, Cantidad, Fecha |
| Evento | idEvento, Detalle, Fecha\_inicio, Fecha\_fin, Ubicacion, Disponibilidad, Promocional, Categoria, Redes\_sociales |
| Reclamo | idReclamo, Titulo, Detalle, Fecha, Tipo\_Reclamo, Estado |
| Respuesta | idRespuesta, Respuesta, Fecha |
| Detalle\_venta | idDetalle\_Venta, Total, Fecha |
| Ticket\_venta | idTicket\_pago, Codigo\_barra, Tipo\_Pago |

Segunda Forma Normal (2FN)

| Usuario | idUsuario, Nombre, Apellido, Correo, Contraseña, FechaRegistro, Estado, Tipo |
| --- | --- |
| Redes\_sociales | idRedes\_sociales, icono, idRed\_Usuario, Url |
| Ticket | idTicket, Titulo, Detalle, Valor, Disponibilidad, Cantidad, Fecha, idEvento |
| Evento | idEvento, Detalle, Fecha\_inicio, Fecha\_fin, Ubicacion, Disponibilidad, Promocional, idUsuario |
| Detalle\_categoria | idEvento , idCategoria |
| Categoria | idCategoria, Nombre |
| Reclamo | idReclamo, Titulo, Detalle, Fecha, Tipo\_Reclamo, Estado, idUsuario |
| Respuesta | idRespuesta, Respuesta, Fecha, idUsuario, idReclamo |

Detalle\_venta idDetalle\_Venta, Total, Fecha, idUsuario, idTipo\_pago Ticket\_venta idTicket\_pago, Codigo\_barra, Tipo\_Pago, idDetalle\_Venta, idTicket, idEvento,

Tipo\_pago idTipo\_Pago, Tipo\_pago

Tercera Forma Normal (3FN)



| Usuario | idUsuario, Nombre, Apellido, Correo, Contraseña, FechaRegistro, Estado, Tipo |
| --- | --- |
| Redes\_sociales | idRedes\_sociales, icono |
| Login\_redes | idRedes\_sociales, idUsuario, idRed\_Usuario |
| Evento\_redes | idRedes\_sociales, idUsuario, Url |
| Ticket | idTicket, Titulo, Detalle, Valor, Disponibilidad, Cantidad, Fecha, idEvento |
| Evento | idEvento, Detalle, Fecha\_inicio, Fecha\_fin, Ubicacion, Disponibilidad, Promocional, idUsuario |
| Categoria | idCategoria, Nombre |
| Detalle\_categori a  Reclamo | idEvento , idCategoria  idReclamo, Titulo, Detalle, Fecha, Tipo\_Reclamo, Estado, idUsuario |
| Respuesta | idRespuesta, Respuesta, Fecha, idUsuario, idReclamo |
| Detalle\_venta  Ticket\_venta | idDetalle\_Venta, Total, Fecha, idUsuario, idTipo\_pago idTicket\_pago, Codigo\_barra, Tipo\_Pago, idDetalle\_Venta, idTicket, idEvento, |

Tipo\_pago idTipo\_Pago, Tipo\_pago

**Modelo Lógico**

****

*****Ilustración 9: Modelo Lógico*

**Modelo Físico**

****

*****Ilustración 10: Modelo Físico*

**Alcances y limitaciones**

****

Alcances

Los aspectos de alcanzaremos en este proyecto son los siguientes:

● Desarrollar una plataforma que permita la recopilación y difusión de una gran variedad de eventos dentro de la región de los lagos que permita de manera gratuita a cada entidad organizadora la opción de registrar, promocionar, administrar y vender entradas de sus eventos a cada uno de los usuarios interesados.

Limitaciones

Las limitaciones detectadas en este proyecto fueron las siguientes:

● Las estrategias creadas en este documento no aplicaran a otras regiones del país.

● La dificultad para encontrar entidades organizadoras para participar de nuestro proyecto

● Posibles demoras en el desarrollo debido a tiempo de aprendizaje y pruebas de nuevos métodos para algunos miembros del equipo

● Posible partida de integrantes del equipo que dificultarían el desarrollo y planificación del proyecto

**Análisis de modelo de desarrollo y metodología a utilizar.**

Durante el desarrollo de nuestro proyecto de software, hay que tener bien definida una buena metodología de desarrollo para realizar las distintas actividades a lo largo del proceso de creación de nuestro producto. Es por eso que es menester realizar una determinación de modelo a seguir y describir las distintas etapas de este modelo. Hay que tener en cuenta que existen varias metodologías para desarrollar un producto de software. Existen algunos como el modelo cascado en el cual se ordenan rigurosamente las etapas del



desarrollo del software, de esto se obtiene que el inicio de una etapa de desarrollo deba de esperar el fin de la etapa anterior.

Otros como el modelo de prototipos que pertenece a los modelos evolutivos, en el cual el prototipo debe de ser construido rápidamente y con la utilización escasa de recursos.

Hay otros modelos más específicos y que han ido evolucionando como el modelo Scrum, que consiste en entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Otro sistema es el análisis y diseño orientado a objetos (ADOO) es un enfoque de la ingeniería de software que modela un sistema como un grupo de objetos que interactúan entre sí. Hay que analizar estas metodologías y elegir de entre todos estos 2 tipos, para después compararlas entre las dos y elegir la metodología más efectiva para trabajar en nuestro proyecto.

Tipos de Modelo de desarrollo

Como mencionamos antes existen varias metodologías de desarrollo de software las cuales nombraremos brevemente:

**Modelo de Desarrollo Orientado a objetos**

Este enfoque representa un domino absoluto en términos de conceptos compuestos por verbos y sustantivos, clasificados de acuerdo a su dependencia funcional. Todo sistema de información requiere de artefactos o componentes (clases) para llevar a cabo tareas. En este método de análisis y diseño se crea un conjunto de modelos utilizando una notación acordada como, por ejemplo, el lenguaje unificado de modelado (UML). Esta metodología aplica técnicas de modelado de objetos para analizar los requerimientos para un contexto (por ejemplo, un sistema de negocio, un conjunto de módulos de software) y para diseñar una solución para mejorar los procesos involucrados.



**Modelo de desarrollo estructurado.**

Técnicas estructuradas de desarrollo de software que utilizan gráficas (diagramas) y textuales (documentos). Este sistema ha ido evolucionando para adaptarse a los sistemas en tiempo real. Posibilita la división de los programas en módulos he introduce el concepto de abstracción. Ejemplo de este tipo de modulo es la cascada.

**Modelo de desarrollo evolutivo**

El desarrollo evolutivo consta del desarrollo de una versión inicial que luego de exponerse se va refinando de acuerdo de los comentarios o nuevos requerimientos por parte del cliente o del usuario final.

Existen dos tipos de desarrollo evolutivo:

● **desarrollo exploratorio:** donde el objetivo del proceso es trabajar con el cliente para explorar sus requerimientos y entregar un sistema final. El desarrollo empieza con las partes del sistema que se comprenden mejor. El sistema evoluciona agregando nuevos atributos propuestos por el cliente.

● **prototipos desechables:** donde el objetivo del proceso de desarrollo evolutivo es comprender los requerimientos del cliente y entonces desarrollar una definición mejorada de los requerimientos para el sistema. El prototipo se centra en experimentar con los requerimientos del cliente que no se comprenden del todo.

**Modelo RAD o de desarrollo de aplicaciones agiles**

Como su nombre lo indica permite la construcción rápida de sistemas utilizables. Está compuesto por un grupo reducido de personas incluyendo desarrolladores y testers del



sistema. También se debe de hacer énfasis al desarrollo de la aplicación cumpliendo correctamente las funcionalidades principales, dejando a un lado a las implementaciones secundarias. Este modelo toma principalmente en cuenta las características de usabilidad, utilidad y rapidez de la ejecución de la aplicación.

**Modelo de Desarrollo SCRUM**

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación. El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint, se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor.

Modelos de Desarrollo a escoger para desarrollar software

Como podemos observar existen varios modelos de desarrollo, cada uno creado para agilizar y mejorar el proceso de desarrollo de software. Obviamente dependiendo del producto a crear el equipo de trabajo debe escoger algún modelo que mejor se adapte a sus necesidades dependiendo de diversas variables (tiempo, capital inicial, equipo de trabajo, tipo de software, etc.). Ahora bien, el equipo de trabajo tiene por objetivo realizar una plataforma para administrar eventos sociales que se realicen en la zona de Puerto Montt. El cliente es una persona que requiere la plataforma para trabajar como una empresa que pueda administrar todos los eventos que se registren en la plataforma en base a lo que el usuario con perfil de organizador quera ingresar.

Teniendo en cuenta que la plataforma web a realizar será responsiva, que pueda operar en cualquier plataforma ya que es una aplicación web y que demás cuente con un sistema de



pago PayPal y tarjeta de débito; el equipo ha pensado en dos modelos de desarrollo para este tipo de software: **Modelo de desarrollo evolutivo o de prototipo** y **Modelo de desarrollo Scrum.**

Desarrollo evolutivo V/S SCRUM

A continuación, se realizará una comparativa entre estos dos modelos de desarrollo explicando su función, propiedades ventajas y desventajas a la hora de utilizarlo en el proyecto de plataforma para organizar eventos:

**Definición**

Como mencionamos antes **El modelo de desarrollo de prototipos o evolutivo** se basa en realizar un prototipo inicial e interactuando con el cliente ir mejorando este diseño hasta llegar al prototipo final, a grandes rasgos.

**El modelo SCRUM**, por su lado; consiste en realizar sprints de 2 a 4 semanas en las cuales se establecen los puntos iniciales a realizar y se utiliza el sprint para cumplir los objetivos establecidos. El modelo SCRUM es más estructurado que el modelo evolutivo puesto que este va estableciendo sus objetivos de desarrollo específicos, en cambio el modelo evolutivo se basa en crear un prototipo que puede que no cuente de objetivos específicos y claros, por lo que el prototipo puede que no funcione correctamente, pero esa es la idea, crear un prototipo para que el cliente vea establezca sus requerimientos funcionales principales.

**Propiedades de cada uno**

Para saber qué modelo escoger hay que conocer sus propiedades básicas y de cómo está estructurado cada uno al utilizarlos.



El Modelo Scrum consta de un proceso que contiene componentes y etapas:

|  |
| --- |

● **Product Backlog:** Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor.

● **Sprint Planning:** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint y organizar cómo lo va a conseguir.

● **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog en una nueva versión del software totalmente operativo.

● **Sprint Backlog:** Lista de las tareas necesarias para llevar a cabo las historias del sprint.



● **Daily sprint meeting:** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada.

● **Demo y retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta los requisitos conseguidos mediante una demonstración del producto. Posteriormente el equipo analiza qué se hizo bien, qué procesos serían mejorables y cómo perfeccionarlos.

También hay que mencionar que este tipo de sistema debe tener roles, para poder desarrollar las actividades dentro de un sprint:

● **Scrum master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología.

● **Product owner (PO):** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio.

● **Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta.

La propiedad del Modelo de prototipos se podría decir que son más simples, Este consta de las siguientes etapas:

● **Recolección y refinamiento de requisitos:** requerimientos funcionales otorgados por el cliente.

● **Modelado, diseño rápido:** se realiza un prototipo Inicial.



● **Construcción del Prototipo:** se desarrolla este prototipo por el equipo.

● **Desarrollo, evaluación del prototipo por el cliente:** el cliente evalúa el prototipo y si este necesita mejoras que se adapten a sus requisitos.

● **Refinamiento del prototipo:** se perfecciona el prototipo en base a las correcciones del cliente.

● **Producto de Ingeniería:** Se obtiene el producto final.

|  |
| --- |

*Ilustración 11: Actividades concurrentes*

El modelo de desarrollo evolutivo es eficaz cuando se cuenta con un equipo de trabajo reducido y se necesita trabajar rapido con los clientes. En el caso ddel grupo actual este sistema es más llamativo a la hora de realizar el proyecto de “plataforma para organizar eventos” debido. Que se cuenta con un grupo reducido y se requiere de un sistema para ir mostrándole al cliente para que sepa bien lo que requiere que la plataforma haga.

**Ventajas y Desventajas**

****

Las ventajas de **Scrum** en cuanto al sistema a realizar podrían ser:

● Cumplimento de expectativas: El cliente establece sus expectativas indicando el valor que le aporta cada requisito.

● Flexibilidad a cambios: Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente.

● 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 ayuda a la obtención de un software de calidad superior.

● Mayor productividad: Se consigue gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.

● Predicciones de tiempos: Mediante esta metodología se conoce la velocidad media del equipo por sprint.

● Reducción de riesgos: permite despejar riesgos eficazmente de manera anticipada. Desventajas:

● **Requiere una exhaustiva definición de las tareas y sus plazos:** Cuando estos dos aspectos no se definen adecuadamente, Scrum se desvanece.

● **Exige una alta cualificación o formación:** No es una modalidad de gestión propia de grupos junior o que apenas están en proceso de formación.

**Ventajas de Modelo Evolutivo**

****

El modelo evolutivo también posee ventajas dependiendo de una determinada situación de desarrollo:

● No modifica el flujo del ciclo de vida: este se mantiene a lo largo del desarrollo. ● Reduce el riesgo de construir productos que no satisfagan las necesidades del cliente.

● Reduce costo y aumenta la probabilidad de éxito: al realizar un prototipo principal se ahorra tiempo en diseño.

● Exige disponer de las herramientas adecuadas.

● Útil cuando el cliente conoce los objetivos generales para el software, pero no identifica los requisitos detallados de entrada, procesamiento o salida. ● También ofrece un mejor enfoque cuando el responsable del desarrollo del software está inseguro de la eficacia de un algoritmo.

**Desventajas**

● Debido a que el usuario ve que el prototipo funciona piensa que este es el producto terminado y no entienden que recién se va a desarrollar el software.

● El desarrollador puede caer en la tentación de ampliar el prototipo para construir el sistema final sin tener en cuenta los compromisos de calidad y mantenimiento que tiene con el cliente.

● Genera mucho tiempo en el desarrollo del sistema.

● Requiere experiencia en la identificación de riesgos.

Analizando las ventajas y desventajas de cada modelo podemos ver que el modelo evolutivo posee mayores ventajas en base al proyecto a realizar, nuestro sistema es un



sistema web, pero poseemos un equipo de trabajo reducido. El cliente es solo una persona por lo tanto no se necesita de mayores equipos de organización. No necesitamos. Como no tenemos una gran formación el Modelo SCRUM sería recomendable para equipos más profesionales y con un sistema para crear más sofisticado que requiera más detalle a la hora de realizarlos. El equipo de trabajo con el que se cuenta es reducido y prácticamente principiante, por lo que un modelo evolutivo en base a fases de creación es más intuitivo y que al crear un prototipo inicial se puede ir adaptando a las necesidades del usuario.

Tabla comparativa

| Nombre  del  Modelo  Modelo  de  Prototipo s | Definición  El modelo  de  desarrollo  de  prototipos o evolutivo se basa en  realizar un prototipo  inicial e  interactuand o con el | Descripción Ventajas  Este diseño  No modifica  conduce a la  el flujo del  construcción de  ciclo de vida.  un prototipo, el  cual es evaluado  Reduce costo  por el cliente  y aumenta la  para una  probabilidad  retroalimentació  de éxito.  n; gracias a ésta  se refinan los  Exige  requisitos del  disponer de  las | Desventajas  Genera  confusiones con el  cliente.  El  desarrollado r puede caer en la  tentación de ampliar el  prototipo. | Aplicaciones  Se utiliza  principalmen te en  proyectos  simples y de equipos  primerizos. |
| --- | --- | --- | --- | --- |



cliente ir mejorando este diseño hasta llegar al prototipo final.

software que se desarrollará.

herramientas adecuadas.

Útil cuando el cliente

conoce los objetivos

generales

para el

software.

Genera

mucho

tiempo en el desarrollo del sistema.

Requiere

experiencia en la

identificació n de riesgos

| Modelo  SCRUM | Consiste en realizar  sprints de 2 a 4 semanas en las  cuales se  establecen  los puntos  iniciales a  realizar y se utiliza el  sprint para  cumplir los objetivos  establecidos . | Se suele  Cumpliment  planificar por  o de  semanas. Al  expectativas:  final de cada  El cliente  Sprint o  establece sus  iteración, se va  expectativas.  revisando el  trabajo validado  Flexibilidad  de la anterior  a cambios.  semana. En  función de esto,  Reducción  se priorizan y  del Time to  planifican las  Market.  actividades en  las que  Mayor  invertiremos  calidad del  nuestros  software.  recursos en el  siguiente Sprint.  Mayor  productivida  d.  Predicciones  de tiempos. | Requiere  una  exhaustiva  definición  de las tareas y sus  plazos.  Exige una  alta  cualificació n o  formación. | Se utiliza  principalmen te en  proyectos a  gran escala y con un grupo de trabajo  mayor, al  nivel de  empresas. |
| --- | --- | --- | --- | --- |



|  |  | Reducción  de riesgos. |  |  |
| --- | --- | --- | --- | --- |

Bibliografía

Modelo de desarrollo:

(https://www.elconspirador.com/2013/08/19/modelos-de-desarrollo-de-software/, s.f.)

Modelo de desarrollo Orientado a objetos:

(es.wikipedia.org, s.f.)

Modelo de desarrollo Estructurado:

(www.goconqr.com, s.f.)



(https://es.wikipedia.org/, s.f.)

Modelo de desarrollo Evolutivo:

● https://ingsoftware.weebly.com/modelo-evolutivo.html

● https://www.ecured.cu/Modelo\_de\_prototipos

● https://es.wikipedia.org/wiki/Modelo\_de\_prototipos

Modelo de desarrollo SCRUM:

● https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum. html

● https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/ proceso-roles-de-scrum.html

● https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum/ proceso-roles-de-scrum.html