***UNIVERSIDAD TECNOLÓGICA NACIONAL***

***FACULTAD REGIONAL CÓRDOBA***

**Ingeniería en Sistemas de Información**

Curso: ***5K4.***

Profesor: ***Ing. Zohil, Julio***

JTP: ***Ing. Aquino, Francisco***

***PROYECTO FINAL***

***Plan de metricas***

Producto: Sistema de Gestión Educativa

Sistema:

***EDUAR 2.0***

Grupo Nro.: 6

INTEGRANTES LEGAJO

***Bazán, María Belén 48071***

***Herrán, Martín Carlos 44633***

***Nicoliello, Pablo Fabián 42318***

***Pastorino, Laura Analía 44647***

*Revisión: 1*

*Última Modificación:*

Historial de Revisión

|  |  |  |  |
| --- | --- | --- | --- |
| **Versión** | **Fecha** | **Descripción del Cambio** | **Autor** |
| 1.0.0 | 24/05/2011 | Version Inicial | Martin Herran |

Tabla de contenido

[1. Contexto 5](#_Toc294025212)

[Sobre el Proyecto 5](#_Toc294025213)

[2. Alcance del Plan 6](#_Toc294025214)

[3. Referencias 6](#_Toc294025215)

[4. Objetivo 6](#_Toc294025216)

[Objetivo de las métricas agiles 6](#_Toc294025217)

[5. Elementos de una Métrica 7](#_Toc294025218)

[1. Métricas 8](#_Toc294025219)

[Sprint BurnDown Chart (SBDC) 8](#_Toc294025220)

[Velocity (Veln) 9](#_Toc294025221)

[Deferred Ratio(DFRT) 10](#_Toc294025222)

[Release BurnDown Chart 11](#_Toc294025223)

[6. Glosario De Términos y Acrónimos 12](#_Toc294025224)

[2. Bibliografía 13](#_Toc294025225)

[3. Anexo 14](#_Toc294025226)

# Contexto

## Sobre el Proyecto

Sea el Proyecto BLPM para el desarrollo del Sistema de Informacion EduAR 2.0 que realiza explotación de información educativa, en la cual existe un proceso de desarrollo de software basado en ScrumUP y define lo siguiente:

* Existe un Rol de Analista Funcional, el cual se encuentra en permanente contacto Product Owner para resolver inquietudes .
* Existe un equipo de arquitectura que en base a los requerimientos efectúa el diseño general de la solución de software.
* Existe un equipo de desarrollo, que basándose en las especificaciones de arquitectura, cumple los alcances y resuelve los defectos en distintos releases.
* Existe un equipo de testing, que basándose en los requerimientos desarrollan distintas suites de test.
* Existe un equipo de implementación de releases.
* Los desarrolladores también deben hacer un Test unitario de la funcionalidad que están desarrollando.
* Por cada falta se debe registrar su nivel de severidad/criticidad, que puede ser:
  + Cosmético: Falta que afecta a la representación visual de la información en el Software.
  + Normal: Una falta que es referida al funcionamiento del software, en el sentido de que puede proporcionar un resultado incorrecto o inconsistente.
  + Urgente: Falta que afecta al normal funcionamiento de una parte del software, pero permite el procesamiento por algún método alternativo.
  + Crítico: Falta que impide la ejecución de todo el sistema o bloquea la utilización de alguno de sus módulos.
* Por último, el entregable, debe pasar por una inspección formal de código, con compañeros del Equipo.

***Defecto****: Errores reportados en la fase de testing, implementación, y los registrados por el cliente.*

***Error:*** *error encontrado en la fase de desarrollo.*

***Faltas:*** *Total de errores, errores más defectos.*

***Defecto rebote:*** *Error generado al corregir otro error.*

***Capacidad individual*** *= dias\_disp \* horas\_diarias\_disp \* porcentaje\_asignacion*

***Capacidad equipo*** *= Suma (capacidad individual)*

# Alcance del Plan

Este documento es parte de los estándares de proyecto para el reporte y seguimiento de Métricas que contribuyan al objetivo del Proyecto y que agreguen valor a la tarea desarrollada. Cada métrica debe tener definido un procesamiento, almacenamiento y si fuera posible, herramientas automatizadas asociadas.

# Referencias

Los siguientes documentos tienen información relevante para el Plan de Métricas.

|  |  |
| --- | --- |
| Identificador del Documento | **Detalles del Documento** |

Informe Preliminar

Plan de Proyecto

# Objetivo

## Objetivo de las métricas agiles

* Se debe tener sólo las métricas realmente necesarias.
* ScrumUp hace mucho énfasis en el control y seguimiento del Proyecto, usando métricas que muestren la evolución y den la posibilidad de proyectar en el tiempo.
* La **velocidad de desarrollo del** equipo, permite:
  + Extrapolar la fecha de finalización del proyecto y/o saber de qué objetivo/requisitos se dispondrá en una fecha determinada.
  + Planificar un nuevo proyecto a partir de la historia anterior.
  + Medir le aprendizaje del equipo, ya que es una métrica que debería aumentar con el tiempo.

# Elementos de una Métrica

* Se debe tener sólo las métricas realmente necesarias.
* Descripción detallada de cada métrica debe incluir los siguientes elementos:

|  |  |
| --- | --- |
| **Elemento** | **Descripción** |
| Nombre de la métrica | Identifica la Métrica |
| Propósito / Objetivo | Describe la métrica y su aplicación. |
| Definiciones de Entrada | Describe cada uno de los parámetros de entrada utilizados en el cálculo de la métrica. |
| Formula | Expresión matemática para calcular la métrica. |
| Responsables de Recolección | Rol del individuo responsables de realizar la recolección de los datos para calcular la métrica. |
| Repositorio de datos | Lugar de almacenamiento de la métrica. |
| Frecuencia | Frecuencia de presentación y análisis de la métrica. |
| Ejemplo de Reporte | Reporte ejemplo de la métrica. Puede ser un grafico ilustrativo para demostrar como presentar la métrica. |
| Unidad de Medida | Unidad de referencia para la métrica. |
| Análisis | Acercamiento al análisis de la métrica. Pueden ser preguntas disparadoras. |

# Métricas

## Sprint BurnDown Chart (SBDC)

|  |  |
| --- | --- |
| **Propósito** | Provee un indicador diario de la velocidad (velocity) del equipo y el progreso respecto del trabajo comprometido para el sprint actual. |
| **Fuentes** | Panel de seguimiento de Sprint Backlog Items. |
| **Elementos de entrada** | Trabajo Pendiente: Cantidad de Sprint Backlog ítems que no fueron comenzadas y esfuerzo restante en las que ya comenzadorn. |
| **Fórmula** | Tendencia de Regresión Lineal  Trabajo Pendiente +  Tendencia = Trabajo pendiente del día anterior a ayer + Trabajo pendiente de ayer + Trabajo pendiente de hoy) / 3 |
| **Responsable** | Scrum Master |
| **Repositorio** | Version One tool |
| **Frecuencia** | Se colecta una vez por día. Es la métrica principal de seguimiento del Sprint. |
| **Unidad de medida** | Esfuerzo |
| **Ejemplo de Reporte** |  |
| **Análisis** | * Cuanto esfuerzo vamos “quemando” en lo que va del Sprint? * Haciendo una proyección, vamos a llegar a cumplir los objetivos del Sprint? |

## Velocity (Veln)

|  |  |
| --- | --- |
| **Propósito** | La Velocity permite a los miembros del equipo conocer el ratio de progreso del equipo completo. |
| **Fuentes** | Minutas de Retrospectivas de Sprint. |
| **Elementos de entrada** | Cantidad de User Stories metidas en cada Sprint |
| **Fórmula** | Vel[n] = # Story Points por Sprint |
| **Responsable** | Scrum Master |
| **Repositorio** | Version One tool |
| **Frecuencia** | Se colecta un punto por Sprint. Se utiliza para la planificación de cada sprint, y para los  ajustes en el release. |
| **Unidad de medida** | Story Points. |
| **Ejemplo de Reporte** |  |
| **Análisis** | * En general, cuantos User Stories desarrollo por Sprint? * A que se debe que en el Sprint N desarrolle menos User Stories que mi promedio de Velocity? |

## Deferred Ratio(DFRT)

|  |  |
| --- | --- |
| **Propósito** | Muestra el número de user stories diferidas por sprint, en relación a la cantidad total de user stories inicialmente comprometidos en el sprint. |
| **Fuentes** | Panel de seguimiento de Product Backlog |
| **Elementos de entrada** | Trabajo Pendiente: Cantidad de Sprint Backlog ítems que no fueron comenzadas y esfuerzo restante en las que ya comenzadorn. |
| **Fórmula** | Deferred Ratio = # User Stories Diferidas/ # Total User Stories del Sprint |
| **Responsable** | Scrum Master |
| **Repositorio** | Version One tool |
| **Frecuencia** | Se colecta un punto por Sprint. Se utiliza para la planificación de cada sprint, y para los ajustes en el release. |
| **Unidad de medida** | Story Points |
| **Ejemplo de Reporte** |  |
| **Análisis** | * Normalmente, cuantos User Stories difiero a siguientes Sprints? * A que se debe que estoy difiriendo? * Problemas en las estimaciones? |

## Release BurnDown Chart

|  |  |
| --- | --- |
| **Propósito** | Ayuda a determinar la probable fecha de finalización del release, teniendo en cuenta la performance (velocidad) del equipo y los cambios introducidos en el alcance. |
| **Fuentes** | Panel de seguimiento de Sprints. |
| **Elementos de entrada** | * Trabajo Pendiente * Trabajos Realizados * Cambios en el Alcance |
| **Fórmula** | Intersección de las líneas de tendencia de la velocidad del equipo y el trabajo introducido al release. Provee una aproximación a la fecha de release. |
| **Responsable** | Scrum Master |
| **Repositorio** | User Stories |
| **Frecuencia** | Se recolecta un punto por Sprint. Se utiliza para la planificación de cada sprint, y para los ajustes en el release. |
| **Unidad de medida** | Esfuerzo |
| **Ejemplo de Reporte** |  |
| **Análisis** | * Como ha impactado cambios de alcance en el compromiso del Proyecto? * Venimos adelantados o atrasados con las fechas? |

# Glosario De Términos y Acrónimos

* Backlog del producto: El Backlog del Producto (o "product backlog") contiene los requerimientos del sistema, expresados como una lista priorizada de elementos del backlog del producto. Esto incluye requerimientos del cliente funcionales y no funcionales, y también requerimientos técnicos generados por el equipo.
* Backlog del sprint: Define el trabajo de un sprint, representado por un conjunto de tareas que deben completarse para cumplir los objetivos del sprint, y por un conjunto de elementos seleccionados del backlog del producto.
* Elemento del backlog del producto: En Scrum, un elemento del backlog del producto ("PBI" o "product backlog item") es una unidad de trabajo lo suficientemente pequeña para que el equipo pueda completarla en un sprint. Los elementos del backlog se descomponen en una o más tareas.
* Estimación ROM (Rough Order of Magnitude): Se utiliza durante el inicio de un proyecto para decisiones estratégicas, planificación a largo plazo o simplemente para responder a la pregunta “aproximadamente cuánto durará el proyecto?”
* Release (Entrega): Es la transición de un incremento potencialmente productivo del producto en algo que pueda ser usado rutinariamente. Las entregas suelen ocurrir cuando uno o más sprints resultan en que el producto tiene suficiente valor como para superar el costo de desplegarlo.
* *RUP: Proceso Unificado de Desarrollo.*
* ScrumMaster: El ScrumMaster es una facilitador para el equipo y para el Dueño del Producto. En vez de gestionar al equipo, el ScrumMaster trabaja para asistir tanto al equipo como al Dueño del Producto,
* Sprint: Una iteración de trabajo durante la cual se implementa un incremento de la funcionalidad del producto. Usualmente, una iteración dura 30 días.
* Stakeholder: Son partes con un interés en el producto bajo desarrollo y/o en el proceso de Scrum.
* Tarea del sprint: En Scrum, una tarea del sprint (o tarea) es una unidad de trabajo generalmente entre 4 y 16 horas. Los miembros del equipo se asignan voluntariamente las tareas. Actualizan la estimación de horas restantes de forma diaria, influenciando así el gráfico de burndown del sprint. Las tareas están contenidas dentro de elementos del backlog.
* User Story: es una representación de un requerimiento de software escrito en una o dos frases utilizando el lenguaje común del usuario y desde su perspectiva.
* PRD: Post Release Defect

# Bibliografía

Introducción a Metodologías Agiles. Apunte de la Cátedra de Métricas de Software

# Anexo