

WORLD CUP PREDICTION MODEL
MLB SOFTWARE

DOCUMENTO DE ESTRATEGIAS DE DESARROLLO

VERSIÓN <1.0>

29/08/2022

TABLA DE CONTENIDOS

World Cup Prediction Model	
MLB Software	1
Documento de Estrategias de Desarrollo	1
Versión <1.0>	1
29/08/2022	1
Tabla de Contenidos	2
Control de Versiones	3
Aprobación de documento	3
Esta versión del Documento de estrategias de desarrollo ha sido aprobada y proveída por los siguientes:	3
Descripción General	4
Organización Capital Humano	4
Requerimientos	5
Especificación requerimientos funcional y no funcional:	5
Analysis y Diseño	5
Desarrollo	6
Proceso de Generación de Software	6
Testing & QA	7
Producción y Mantenimiento.	7

CONTROL DE VERSIONES

Versión	Fecha revisión	Cambios
[1.0]	August 29th 2022	Document creation

APROBACIÓN DE DOCUMENTO

Esta versión del Documento de estrategias de desarrollo ha sido aprobada y proveída por los siguientes:

Nombre	Título	Fecha de Aprobación	Firma
Luis Larios	Administrador	4/9/2022	Aprovado
Brian Salomons	Documentacion	4/9/2022	Aprovado

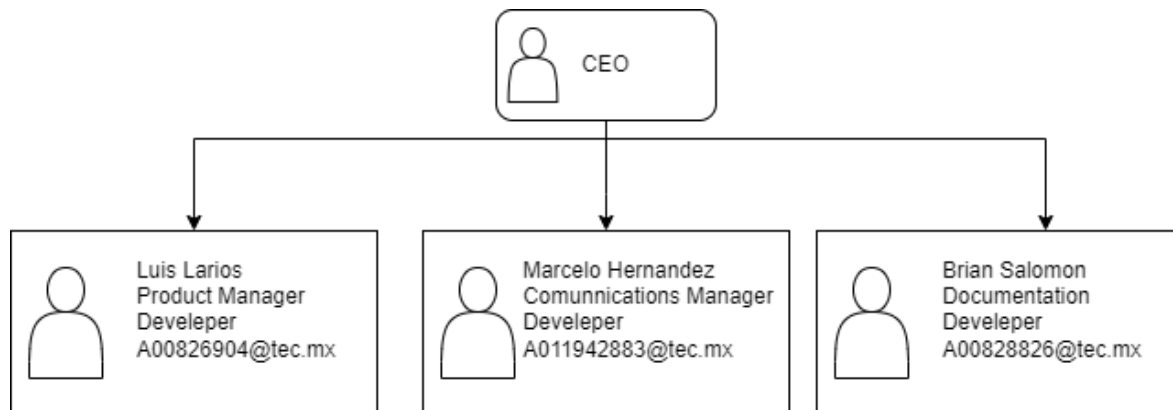
DESCRIPCIÓN GENERAL

Este documento detalla la metodología y estrategias de desarrollo que se usarán durante el ciclo de vida del proyecto “World Cup Prediction Model” para nuestro cliente Wizeline. Este Documento describe los requerimientos, el análisis, el diseño, construcción, QA, producción, y mantenimiento durante el desarrollo del proyecto.

En esta definición se identifican las personas que tienen interés o estarán vinculados de alguna forma en el resultado del proyecto, proceso o componente de software.

Finalmente se definen los canales de comunicación válidos para el desarrollo del proyecto, proceso o componente de software.

ORGANIZACIÓN CAPITAL HUMANO



REQUERIMIENTOS

Especificación requerimientos funcional y no funcional:

Para este proyecto se establecieron los requerimientos basándonos en una metodología ágil, iterativa, con el fin de poder seguir iterando/agregando requerimientos a nuestra documentación conforme a la comunicación que tenemos con nuestro cliente Wlzeline.

Esta decisión fue tomada debido a la baja comunicación que se tuvo con el socio formador. Dentro de nuestra organización se repartió el rol de entrevistador quien tuvo el deber de acercarse a nuestro cliente y ayudarlos a establecer los requisitos del sistema. Nuestro entrevistador tiene la misión de guiar y validar el trabajo de desarrollo por medio de nuestra organización y nuestro cliente.

ANALYSIS Y DISEÑO

Para el análisis y diseño, continuamos usando la metodología de prototipado, ya que es iterativa, y nos permite construir por encima de nuestros avances.

Este modelo nos permite continuar avanzando en diferentes áreas de la aplicación web, sin la necesidad de empezar desde 0 con cada cambio, de el socio formador.

La desventaja de este modelo es que abarca una amplia cantidad de componentes, durante el desarrollo, por lo que puede darse errores durante el desarrollo al querer concentrarnos en la responsividad en vez del funcionamiento establecido en nuestro documento de requerimientos.

DESARROLLO

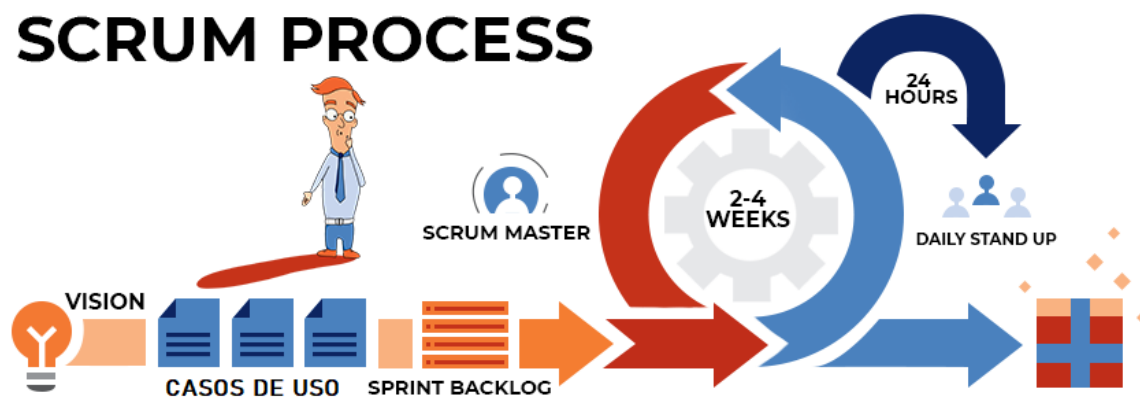
Una vez se desarrollan los documentos y son validados por nuestro Cliente y nuestro equipo de consultores externos (Profesores) en la fase de Requisitos funcionales y no funcionales se designa, se designará a un Arquitecto de software el cual a partir de los casos de uso establece el Diseño de la solución.

Proceso de Generación de Software

Para el desarrollo de software se usará principalmente React como framework para la aplicación web y Python para el desarrollo del modelo predictivo.

Proceso Incremental E Iterativo.

El desarrollo incremental e iterativo (Ágil) es una de las estrategias que mejor se acoplan al desarrollo de software definido en este proyecto. La especificación funcional se captura mediante casos de uso y son los que dirigen el desarrollo. La idea es que cada sprint tome un conjunto de Casos de Uso y transite el flujo definido en cada una de las iteraciones relacionadas a la codificación: análisis, diseño, implementación , prueba y despliegue.



La idea de usar SCRUM es que es rápido y nos permite organizar el proyecto y nuestro tiempo de manera más eficiente ya que tenemos un backlog donde pondremos de manera prioritario las utilidades que buscamos implementar esta iteración del proyecto. Otra ventaja es su flexibilidad, podemos arreglar fallos y faltas durante la etapa de desarrollo.

La desventaja de scrum es que requiere gran disciplina del equipo y las faltas en la organización de los sprints tienden a retrasar los procesos.

La comunicación se puede dar por medio de mensajes instantáneos y redes sociales

TESTING & QA

Una vez se tiene una versión del sistema programada a un nivel comparable al Caso de USO, se procederá a implementar dicho software en los servidores designados para pruebas de solución.

Unit testing: En este tipo de pruebas se busca probar las partes más pequeñas de la aplicación, a los que llamaremos unidades, estas serán privadas individualmente e independiente una de otras, para probar su propia operación.

Pruebas de Integración: Se refiere a las pruebas en la que diferentes unidades, módulos o componentes del software desarrollado serán privados para que funcionen juntos como una sola entidad.

Pruebas de usuarios: Consiste en aquellas actividades de validación realizadas por los usuarios finales sobre el software desarrollado en los ambientes de pruebas con la finalidad de verificar el cumplimiento de la funcionalidad detallada en los casos de uso vs el componente de software producido.

PRODUCCIÓN Y MANTENIMIENTO.

La etapa final final, de las metodología de SCRUM es la etapa en la que el cliente emplea el código desarrollado, por lo que es esencial tener comunicación con nuestro socio. Con la intención de ofrecer mantenimiento al producto. Este puede consistir de correcciones a funcionalidades existentes y/o agregar funcionalidades.

Esta es una de las ventajas de usar procesos ágiles como lo son scrum, ya que no se tiene que empezar desde cero para agregar funcionalidades al sistema y cumplir con las necesidades de negocio del cliente.

La desventaja es que por la naturaleza acelerada de SCRUM, se tiende a otorgar menos tiempo a la documentación de estos cambios y mantenimiento.