

DEVOPS

Fundamentos de la nube

CHRISTIAN ADRIÁN CABRAL PENICHE

INGENIERÍA EN DESARROLLO DE
TECNOLOGÍA Y SOFTWARE

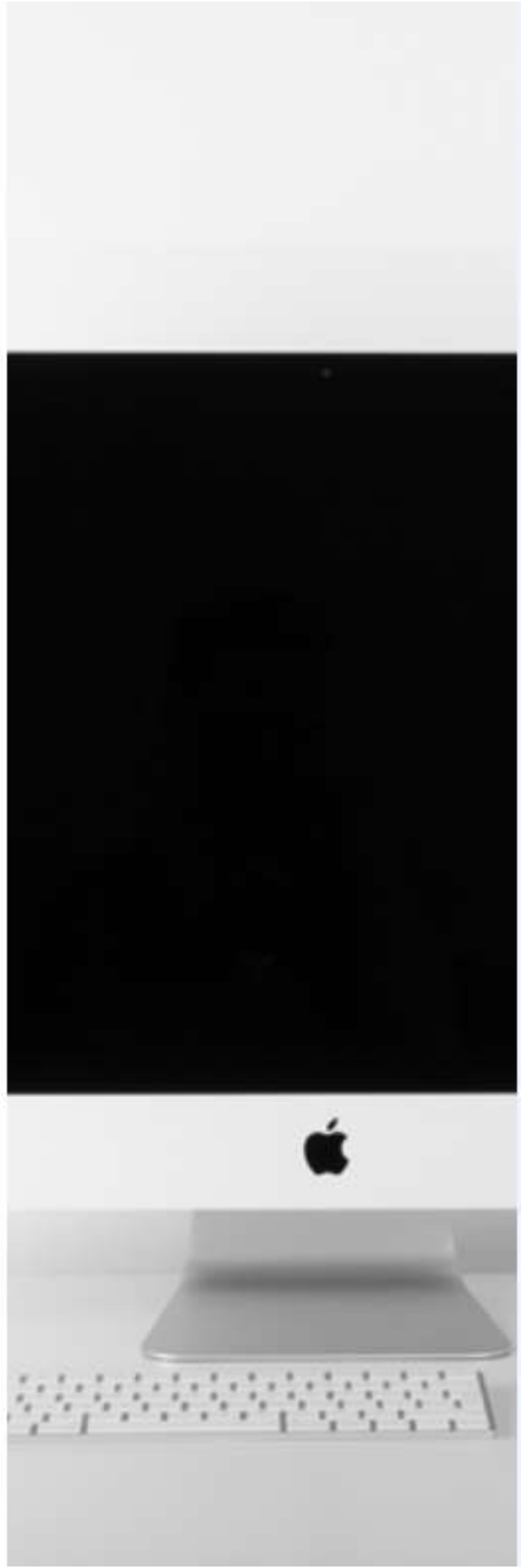
CUARTO SEMESTRE

27 DE FEBRERO DE 2020



TABLA DE CONTENIDO

03	INTRODUCCIÓN
06	METODOLOGÍA SCRUM
13	SISTEMA DE SOPORTE Y SEGUIMIENTO DE TICKETS
15	FLUJO DE TRABAJO DE CONTROL DE VERSIONES



Introducción

03





Introducción

Para garantizar el éxito y el correcto funcionamiento de este proyecto, se utilizará la metodología de desarrollo de software DevOps, la cual comprende un conjunto de filosofías, prácticas y herramientas que permiten el desarrollar aplicaciones a una mayor velocidad y eficiencia que las metodologías tradicionales. Los miembros del equipo tendrán que utilizar distintos programas para automatizar procesos que antiguamente se hacían de forma manual, con el objetivo de trabajar de una manera más rápida y eficaz.



La metodología de desarrollo ágil Scrum será vital para el desarrollo de este proyecto, debido a que distintas de sus prácticas serán utilizadas por el equipo de trabajo. Una de ellas es el uso de los Sprints, intervalos de tiempo con un objetivo a conseguir y un producto a entregar en un período establecido de tiempo. De igual manera, reuniones cada cierto tiempo establecido tendrán que ser hechas por los miembros involucrados, como el Daily Scrum (reunión diaria) y el Biweekly Scrum (reunión cada dos semanas). Estas reuniones se realizarán con los objetivos de que todos los integrantes del equipo estén enterados de los que están haciendo los demás y de resolver los problemas de la forma más rápida y eficiente posible.



En este documento, se describirán y mostrarán las distintas aplicaciones del proceso de DevOps que serán utilizadas por el equipo de trabajo: en el área de planificación, la herramienta a utilizar será Trello; en cuanto al soporte y seguimiento de tickets, se usará el software FreshDesk; en cuanto al flujo de trabajo y control de versiones, se utilizará GitHub por medio de la aplicación Fork.



Metodología Scrum



06

Sprints



En este proyecto, por medio de la metodología Scrum, se trabajará por medio de Sprints: ciclos o intervalos con dificultad, tiempos e integrantes ya establecidos. Los Sprints tendrán que tener una duración máxima de tres semanas y cumplir los objetivos específicos ya determinados en su planeación.

Cada Sprint debe tener una meta de lo que tiene que ser entregado, dando como resultado un incremento de producto. El tiempo que se debe de invertir en cada intervalo se define por medio de la dificultad de las tareas a realizar, el número de integrantes involucrados y los avances que se vayan dando en el proyecto.

Es importante recalcar, que a diferencia de las metodologías tradicionales, el equipo de trabajo también se debe de involucrar en la creación y el desarrollo de los intervalos de trabajo (a diferencia de un sistema antiguo en donde el jefe simplemente le ordena a los empleados), al tomar en cuenta la opinión y las actualizaciones diarias en el Daily Scrum, por lo que en caso de surgir un problema, de necesitar modificar el intervalo de tiempo o de requerir ayuda adicional, los integrantes del equipo puedan apoyarse entre si para llevar a cabo el trabajo.

Después de la conclusión de cada Sprint, un nuevo Sprint empezará de forma inmediata. En este sistema, no importa que algún miembro del equipo termine antes el trabajo del Sprint, por el contrario, favorece el flujo de trabajo del proyecto y permite que la persona siga trabajando en otras partes donde pueda ser de utilidad. Esta práctica permitirá que el software, en ciertos casos, se pueda entregar antes al cliente, lo que en muchos casos es recompensado.



Scrum Daily

Con el objetivo de inspeccionar y actualizarse en los avances realizados durante el día anterior, cada día durante el Sprint se realizará una reunión de 15 minutos máximo, llamada Daily Scrum. En ella, los miembros del equipo informarán sus avances y reportarán cualquier problema o asunto por resolver.

En el Daily Scrum, los integrantes del equipo sincronizarán sus actividades y crearán un plan para las próximas 24 horas. De preferencia, las reuniones deben de ser siempre a la misma hora, en este caso se darán en la mañana antes de empezar a trabajar, a las 8 a.m.

Las personas del equipo le tendrán que contestar tres preguntas al Scrum Master:

-¿Qué hiciste ayer?

Al contestar esta pregunta, cada miembro del equipo de desarrollo explicará cuáles fueron los avances que realizó el día anterior, con el objetivo de que todas las personas estén al tanto de los progresos de los demás. Los progresos podrán ser verificados mediante el servicio de repositorios GitHub.

-¿Qué harás hoy?

Los integrantes informarán cuáles son las tareas que tienen que realizar en el día, según los objetivos y tiempos ya establecidos anteriormente. De esta manera, todos los miembros conocerán lo que harán los demás en el día.

-¿Hay algún problema?

Los miembros del equipo tendrán que informar de todos los problemas y/o atrasos que hayan surgido el día anterior y que no permitieron que el objetivo para ese día se cumpla a tiempo. El asunto es que todo el equipo aporte a encontrar una solución o que los tiempos para la entrega sean modificados.





Scrum Biweekly

Cada dos semanas, una reunión más grande se llevará a cabo entre los miembros del equipo para revisar los avances hasta la fecha, los problemas que hayan surgido y de checar si los tiempos están siendo respetados. En ella, se verificarán los objetivos cumplidos durante el periodo y se buscará solución a todos los problemas que no se hayan podido resolver al momento.

El Biweekly Scrum se llevará a cabo a la misma hora que los Daily Scrum (8 a.m.), la duración de la reunión será de alrededor de dos horas y todas las personas del equipo de desarrollo, el Product Owner y el Scrum Master formarán parte de ella.





Herramienta Scrum

En este proyecto se utilizará el software de gestión de proyectos Trello, una herramienta SaaS que será bastante útil al momento de gestionar la colaboración de los distintos miembros del equipo y de manejar los objetivos a cumplir de cada Sprint. En esta aplicación se podrá verificar en lo que está trabajando cada integrante, el progreso que lleva y el tiempo que le queda para concluir la actividad.

Es importante recalcar que a pesar de que el software Trello esté basado en el sistema Kanban, también nos puede ser bastante útil para implementarlo por medio de la metodología Scrum, realizando unas cuentas modificaciones. Al utilizar Scrum, cada una de las tarjetas del proceso de Kanban se vuelve un Sprint.



Algunas alternativas a Trello para implementar la metodología Scrum son: Atlassian JIRA, VersionOne, CA Agile Solutions, TargetProcess, Active Collab, Pivotal Tracker y Taiga.



ActiveCollab



targetprocess
SEE. CHANGE



PivotalTracker

Ejemplo

Backlog SCRUM Personal Privado Invitar

Planeación

- Revisión de tareas para desarrollo
🕒 28 de feb. 📊 0/2
- Definición de objetivos de producto
🕒 23 de feb. 📊 0/3
- Definición de lenguajes a desarrollar en plataformas
🕒 25 de feb. 📊 0/2
- Generación de lineamientos de calidad
🕒 4 de mar. 📊 0/2
- Generación de plan de comunicación
🕒 23 de feb. 📊 0/4
- Generación de gastos fijos y variables
🕒 23 de feb. 📊 0/3
- Creación de diccionario de datos
🕒 4 de mar. 📊 0/3
- + Añada otra tarjeta

Diseño

- Branding (Logo, imagen corporativa)
🕒 23 de feb. 📊 0/3
- + Añada otra tarjeta

Base de datos

- Definición de gestor de base de datos
🕒 4 de mar. 📊 0/3
- Generación de documento de lineamientos de base de datos
🕒 9 de mar. 📊 0/2
- + Añada otra tarjeta

Documentación

- Generación de documento de estándares de código
🕒 4 de mar. 📊 0/2
- Documentación de procesos
🕒 9 de mar. 📊 0/4
- + Añada otra tarjeta

Web

- Diseño UI para sitio web
🕒 17 de mar. 📊 0/3
- Programación Front End sitio web
🕒 1 de abr. 📊 0/6
- Programación BackEnd sitio web
🕒 23 de mar. 📊 0/5
- Testing sitio web
🕒 10 de abr. 📊 0/4
- + Añada otra tarjeta

Móvil

- Diseño UI app IOS/ANDROID
🕒 10 de mar. 📊 0/4
- Programación app IOS
🕒 25 de mar. 📊 0/3
- Programación app ANDROID
🕒 23 de mar. 📊 0/5
- Testing IOS
🕒 10 de abr. 📊 0/3
- Testing ANDROID
🕒 10 de abr. 📊 0/3
- + Añada otra tarjeta

QA

- Generación de matriz de pruebas
🕒 24 de abr. 📊 0/4
- Proceso QA sitio
🕒 24 de abr. 📊 0/3
- Proceso QA app IOS
🕒 24 de abr. 📊 0/3
- Proceso QA app ANDROID
🕒 24 de abr. 📊 0/4
- + Añada otra tarjeta

Servicios de la nube

- Montaje arquitectura en nube producción
🕒 24 de mar. 📊 0/3
- Montaje arquitectura en nube testing
🕒 24 de abr. 📊 0/3
- + Añada otra tarjeta

Roles de trabajo



Equipo web:

- Ana Castillo
- Hanna Lizarraga

Equipo móvil:

- Cristell Naranjo
- Manuel Canul
- Christian Cabral

Equipo Quality Assurance:

- Hanna Lizarraga
- Ana Castillo
- Manuel Canul
- Christian Cabral

Servicios de la Nube:

- Cristell Naranjo

Diseño de Base de Datos:

- Ana Castillo

Documentadores:

- Manuel Canul
- Christian Cabral

Diseño:

- Ana Castillo

Project Manager:

- Alfredo Bolio



Sistema de soporte y seguimiento de tickets



13

Sistema de tickets para soporte y seguimiento

El software de soporte y seguimiento de tickets a utilizar en el proyecto será Freshdesk, una herramienta SaaS que servirá para llevar el control de las quejas, sugerencias y comentarios de los usuarios, con el objetivo de hacer más fácil, rápido y efectivo el proceso de peticiones de las personas.

Al proporcionar un punto de contacto a los usuarios para que manden sus comentarios a resolver por medio de un correo electrónico establecido y organizar los tickets en distintas categorías (por ejemplo: asignado, no asignado, vencido, sin resolver, en espera, vence hoy, etc.), los problemas relacionados con el software del proyecto podrán ser atendidos de una manera mucho más ordenada, rápida y eficaz a diferencia de hacerlo de forma manual.



Freshdesk proporcionará distintas características y funcionalidades que serán bastante útiles para atender las problemáticas del proyecto; incluso, es capaz de proveer soporte en bastantes plataformas, como pueden ser: e-mail, teléfono, chat, Twitter, Facebook y bots.

Algunas alternativas de sistemas de tickets de seguimiento a Freshdesk son: Vision Helpdesk, Zoho Desk, HelpDesk, SpiraTest y Zendesk.



15

Flujo de trabajo de control de versiones





Flujo de trabajo de control de versiones

Para el flujo de trabajo de control de versiones, se utilizará el servicio de desarrollo colaborativo para proyectos GitHub, el cual utiliza el sistema de control de versiones Git. GitHub proveerá al proyecto de una interfaz gráfica web para complementar a Git.

De igual manera, se utilizará la aplicación Fork via GitHub para proveer de una interfaz mucho más amigable, gráfica y útil para facilitar los cambios y versiones del desarrollo del proyecto. Fork permitirá a los miembros del equipo de desarrollo bastantes características útiles, como commit, fetch, pull, push y resolver conflictos.

Fork funcionará utilizando un repositorio de GitHub, con el objetivo de probar y comparar distintas versiones del software, sin preocuparse de que alguna parte del código se pierda y no se pueda recuperar. Esto permitirá el eficiente desarrollo del programa entre los distintos miembros del equipo, al ser capaz de subir, bajar y guardar los cambios en tiempo real y de poder lidiar con los errores que puedan surgir en el proceso.

Algunas alternativas a Fork pueden ser los siguientes clientes para Git: Github Desktop, SourceTree, GitKraken, SmartGit y Git Cola. De igual manera, el servicio de alojamiento web Bitbucket es una gran alternativa a GitHub, es capaz de funcionar con los sistemas de control de versiones Git y Mercurial.



SmartGit



Bitbucket

Ejemplo

File Repository View About Fork 1.45.0.0

Quick Launch Fetch Pull Push Stash

documentoDevOps detached HEAD

New Branch Console Theme Orientation Feedback Search

documentoDevOps

Changes

All Commits

Filter

Branches

- Dev
- master

Remotes

Tags

Stashes

Submodules

Programación app ANDROID

Diseño UI para sitio web

Programación BackEnd sitio web

Dev Proceso QA sitio

master Testing sitio web

Programación Front End Sitio Web

ChristianCabra199	64ec915	20 Feb 2020 23:34		
ChristianCabra199	19e5920	20 Feb 2020 23:33		
ChristianCabra199	2ff74c7	20 Feb 2020 23:30		
ChristianCabra199	7c1cb8f	20 Feb 2020 23:26		
ChristianCabra199	0d3ff35	20 Feb 2020 23:23		
ChristianCabra199	31a5435	20 Feb 2020 23:22		

Commit Changes File Tree

ChristianCabra199 19e5920 20 feb, 2020 23:33 Diseño UI para sitio web

index.html

```
1 <html>
2 <body>
3
4 HOLA CLASE, PRIMERA PAGINA web.
5 <br>
6 NUEVO MENSAJE
7 <br>
8 CAMBIO EN MASTER 2
9
10 </body>
11 </html>
```

