Primera entrega. Equipo 4

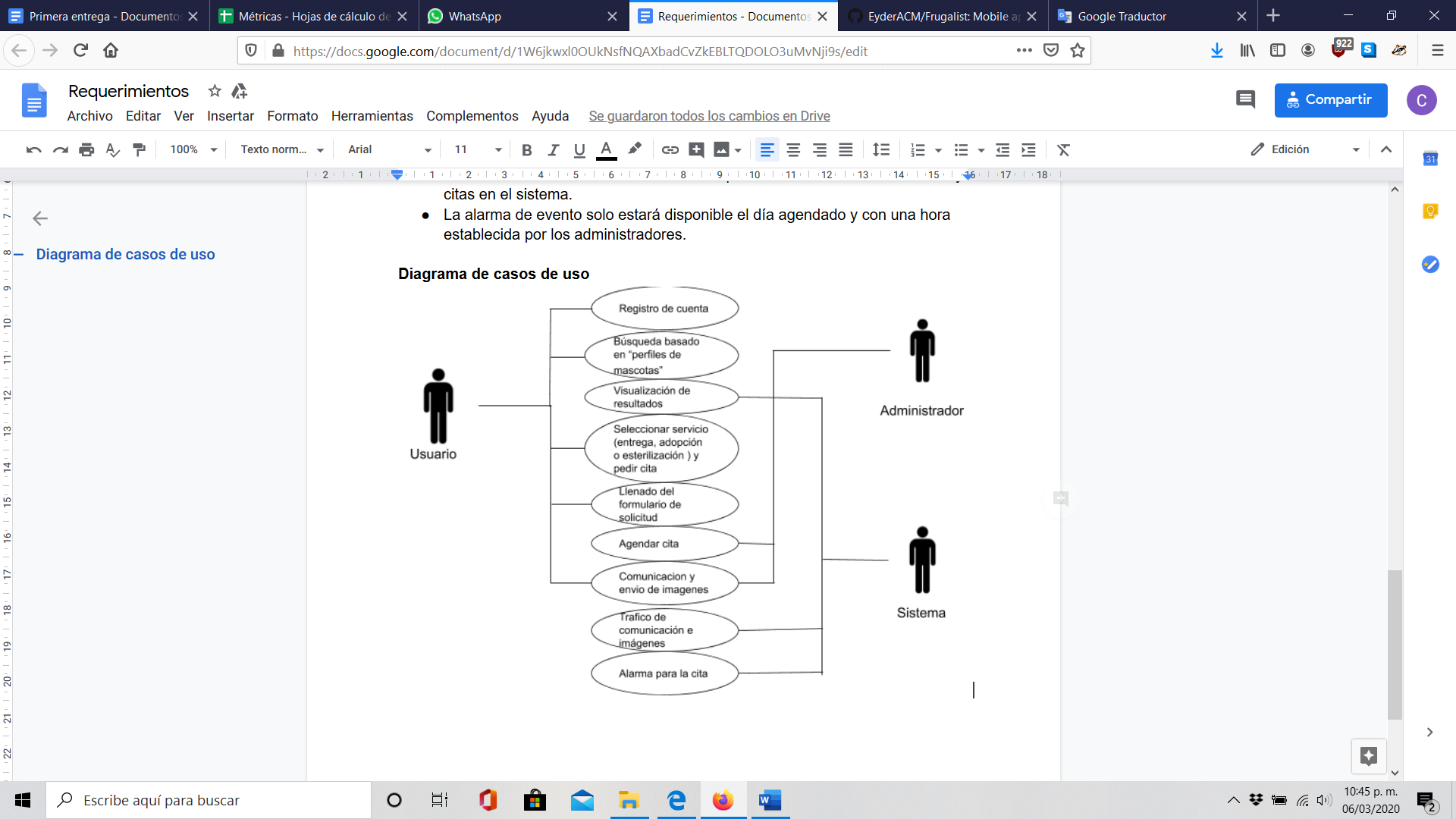
1. Definición inicial de requerimientos.
   1. Actores del sistema

Administrador: Persona miembro de la ONG “Albergue Franciscano Del Animal Desprotegido”

* Puede agendar actividades, sesiones o eventos
* Puede programar el día y la hora para una cita con el usuario que lo solicitó
* Puede comunicarse con el usuario

Usuario: Persona que utiliza los servicios de la ONG

* Puede solicitar una cita con un miembro de la ONG para algún servicio (entrega, adopción o esterilización de mascota)
* Puede hacer una búsqueda en el sistema basado en los elementos de los “perfiles de las mascotas”, de acuerdo con la selección de criterios que haya aplicado
* Puede comunicarse con los administradores
  1. Requerimientos del usuario
* El usuario debe registrar una cuenta introduciendo nombre, nombre de usuario, correo, contraseña, número telefónico, dirección, como ejemplo.
* El usuario podrá buscar y visualizar el resultado de la búsqueda basado en los elementos de los “perfiles de las mascotas”.
* El usuario podrá seleccionar unos de los servicios de la ONG (adopción, entrega o esterilización de algún animal) y solicitar una cita en el albergue.
* El usuario podrá comunicarse con los administradores.
* El usuario podrá mandar fotos e imágenes en un formato específico a los administradores.
* El usuario podrá rellenar el formato de solicitud con los datos correctos requeridos.
  1. Requerimientos del sistema
     1. Funcionales
* El sistema debe dar acceso a un registro de usuario.
* Hace una búsqueda simple o combinaría por características dentro el espectro de selección otorgado por el sistema.
* El sistema permitirá el tráfico de mensajes entre los administradores y usuarios.
* El sistema permitirá el tráfico de fotos e imágenes en el formato establecido por el mismo, entre usuarios y administradores.
* El sistema permitirá la calendarización de citas y eventos creados por los administradores.
* El sistema generará una alarma a ambos tipos de usuarios al llegar la fecha agendada.
* El sistema permitirá a los administradores la calendarización de citas y eventos en una fecha, de tal manera que varios puedan suceder el mismo día y al mismo tiempo.
  + 1. No funcionales
* El registro de usuario para tener acceso al uso del sistema.
* El registro será distinto para administrador y usuario general.
* Para una búsqueda sin una selección de características el sistema arrojará todas la entradas registradas en este.
* Los mensajes tendrán una limitación máxima de 1000 caracteres por cada uno.
* Únicamente el usuario administrador podrá generar y enviar solicitudes.
* El tipo de mensaje solicitud solo podrá ser rellenado con los datos requeridos, aunque estos serán comprobados por los administradores para su correcto uso.
* El sistema generará una marca de agua para todas las fotos enviadas a través de este, al ser mandadas a los administradores.
* Únicamente los usuarios administradores podrán generar entradas, eventos y citas en el sistema.
* Únicamente los usuarios administradores podrán eliminar entradas eventos y citas en el sistema.
* La alarma de evento solo estará disponible el día agendado y con una hora establecida por los administradores.
  1. Diagrama de casos de uso



* 1. Entidades de información

La principal entidad de información será la misma ONG “Albergue Franciscano Del Animal Desprotegido”, que nos brindará la información acerca de los servicios que brindan y sus procedimientos.

Y también se utilizará la guía de desarrollo de Dart por Google que nos servirá de orientación para la codificación del sistema.

1. Definición de estándares de codificación

El lenguaje de programación que se utilizará para el desarrollo del sistema será Dart con Flutter, el estándar de codificación será lo siguiente:

Todo el nombrado de archivos, funciones y clases se utilizará el método de CamelCase.

* 1. Plantilla para la documentación

Se utilizará HTML como plantilla de documentación con la ayuda de la herramienta de DartDoc que se explicará en el siguiente punto.

* 1. Herramienta para la generación automática de documentación

La herramienta de documentación que se utilizará es el software DartDoc, que generará automáticamente la documentación en HTML.

1. Proceso de desarrollo.
   1. Métodos y herramientas

-Diseño: se utilizará el software Figma para crear un modelo de interfaz y así, dar una idea general de la apariencia del sistema.

Construcción: se utilizará VS Code para la codificación con el lenguaje Dart; también se utilizará Flutter que servirá para el desarrollo del sistema; Genymotion para emular el dispositivo móvil; y GitHub Desktop para el control de versiones.

-Documentación: Se utilizará el software DartDoc que generará la documentación en HTML.

* 1. Comunicación

Para una comunicación más ágil y eficiente se crearán diversos canales en la plataforma Slack. Cada uno contendrá un tema específico en el cual se trabajará.

1. Programación

En este canal se tratarán los tópicos relacionados a la programación así como al lenguaje a usar, dudas y distribución de los códigos.

1. Scrum

Apartado dedicado al proceso Scrum usado por el equipo; las dudas y avances de los distintos roles asignados.

1. Pruebas

En esta sección se reportan los resultados de las pruebas realizadas al sistema, los fallos y los puntos a optimizar.

* 1. Planeación

Para la realización del proyecto, se utilizará el método de Scrum, en lo cual consiste el seguimiento de los siguientes pasos:

1. Product Backlog

En el ese paso el equipo se reunirá para la conceptualización del sistema y los requerimientos de este y cuales serán priorizados. El equipo guardará los requerimientos priorizados en una lista donde se definirá la fecha de inicio y término de cada uno.

1. Sprint planning

El equipo se reunirá para planificar qué tareas del requerimiento realizará cada integrante para el sprint.

1. Sprint

El equipo trabajará en el desarrollo de las tareas definidos en el product backlog, cada sprint durará 2 semanas. Cada día el equipo hace un daily scrum, donde cada integrante dirá lo que hizo ayer, lo que hará y de algún problema que tengan.

1. Sprint review

Al término del sprint, el equipo se reunirá para discutir y revisar los avances que se han logrado en el sprint.

1. Sprint retrospective

El equipo se reúne de nuevo con el objetivo de analizar la forma en que se trabajó en el sprint anterior y discutir en qué se puede mejorar para el siguiente iteración

* 1. Monitoreo

El monitoreo del equipo estará representado dentro del Daily Scrum para inspeccionar el progreso hacia la meta de Sprint y llegar al objetivo dentro de este.

Entre cada Daily Scrum, se llevará a cabo un Scrum Taskboard con una lista de objetivos para completar las iteraciones (Product Backlog Items),esto actuará como un radiador de información a todo el equipo y estar enfocados en el objetivo diario.

De igual modo se hará monitoreos correspondiente a cada Sprint review y Sprint retrospective.

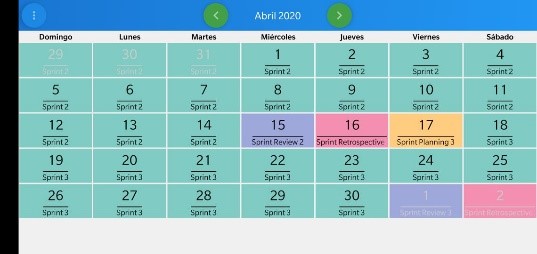
* 1. Bitácora

A lo largo de cada Daily Scrum, Sprint retrospective y Sprint Review se tomará un registro en el que se anotarán de forma cronológica todas las actividades que se deben o se llevan a cabo; las soluciones, avances, observaciones, posibles obstáculos y resultados.

* 1. Métricas

La métrica contiene las actividades, las fechas de inicio y termino, el tiempo para terminarlas, la persona encargada, el porcentaje de aportación y estado. Para medir el porcentaje se planteó la siguiente formula: <https://docs.google.com/spreadsheets/d/1wTUxFFNAdjdoxyD1S7fWjaNvPc0PBD_wMZlKD-DoViU/edit#gid=2066365780>

* 1. Calendario de actividades



* 1. Repositorio

<https://github.com/CarlosGreene/ProyectoEquipo4>