## Introducción:

## 1.0 ¿Qué es *logrolling*?

*Logrolling* es un concepto nacido en Norteamérica que se define como “La acción de intercambiar favores”, sobre todo aplicado en un contexto político. Nosotros queremos extrapolar la idea a la ayuda que pueden ofrecer diferentes integrantes de una misma comunidad.

## Propósito del plan

Cada vez más españoles sufren ansiedad, y el estrés constituye una de las principales causas de bajas laborales en España, el país europeo con más estrés en el entorno de trabajo. La principal causa del estrés es el exceso de actividades y la falta de tiempo, por lo que no nos sorprende que quienes más sufren este trastorno son las parejas con hijos y los estudiantes entre otros grupos.

Con el objetivo de solucionar este problema vamos a desarrollar una aplicación que permita poner en contacto a diferentes integrantes de una comunidad con la intención de que cada usuario pueda ofrecer sus habilidades y su tiempo así como demandar el de otro, en forma de favores. Consideramos que nuestro proyecto no solo ayuda a solucionar este problema social, sino que contribuye a crear un sentimiento de comunidad entre los usuarios que utilicen la aplicación.

Entre el tipo de favores que incluirá la aplicación se encuentra la realización de recados, transporte de individuos, favores particulares (ej: cuidado de mascota) y muchos otros.

## 1.2 Propósito del plan

Nuestro proyecto está orientado a la obtención de diferentes recompensas por la realización de distintos favores publicados por otros usuarios. Por ello la aplicación está destinada a un público general, en especial aquellas personas que puedan ofrecer parte de su tiempo para ayudar a otras personas, y a aquellas que necesitadas de tiempo por el exceso de actividades en su día a día.

## 1.2.1 Funciones principales

|  |  |  |
| --- | --- | --- |
| Módulo | Función | Descripción |
| Usuario | Registro | En un primer instante se le dará la opción al usuario de registrarse en la aplicación. |
| Log-in y log-out | El usuario podrá iniciar y cerrar sesión cuando quiera (aunque esta se mantendrá abierta por defecto) |
| Favores | Publicar un favor/favor múltiple | Un usuario podrá publicar un favor (una necesidad) ofreciendo una retribución (en *grollis*) estandarizada con el objetivo de que otro usuario lo realice. |
| Ofrecer un favor/favor múltiple | Un usuario puede ofrecer un favor (una habilidad específica o un recado normal) esperando una compensación concreta (en *grollis*). A la hora de realizar un recado como “hacer la compra” un usuario podrá establecer la opción de que no solo uno sino varios soliciten su ayuda. |
| Realizar favor | Un usuario puede adjudicarse un favor con la motivación de obtener una retribución (en grollis) tras su realización. |
| Atribuirse favor | Un usuario que necesite un favor ya ofrecido podrá reservarlo, para llevarlo a cabo. |
| Chat | Los usuarios se podrán comunicar a través de un chat privado para concretar detalles del favor a realizar y la forma. |
| Premios y compras | Compra de *grollis* | Los usuarios podrán pagar en la aplicación para recibir *grollis* a cambio. |
| Intercambiar *grollis* por premios | Si el usuario tiene suficientes unidades de la moneda virtual podrá intercambiarlas por premios de amazon. |
| Buscador y geolocalización | Buscar favores | Un usuario puede buscar favores por su nombre. |
| Buscar favores cerca | Un usuario puede buscar favores por distancia a su localización. |

La aplicación tiene diferentes funcionalidades las cuales hemos dividido en 4 módulos. Cualquier usuario con la aplicación podrá llevar acabo estas operaciones.

## 1.2.2 Aspectos de rendimiento

Tenemos la intención de que nuestra app tenga un mercado nacional en un principio sin descartar el objetivo internacional, además el público objetivo es amplio. Por esta razón, el servidor va a tener que ser capaz de almacenar la información de una cantidad superior a los 100.000 usuarios (al menos en un comienzo) en una base de datos cuyo manejo tiene que estar organizado de manera que el registro de varios usuarios o el acceso simultáneo de varios de ellos no aumente el tiempo de respuesta de la aplicación.

Relacionado con el tema de pagos, es importante el aspecto de la seguridad a la hora de realizar transacciones para obtener tiempo (grollis) a cambio de dinero.En relación a la moneda virtual (grollis) una función importante de la aplicación es la obtención de recompensa, ello depende de la conexión con los servidores de Amazon.

Por último, uno de los principales aspectos a tener en cuenta es que la aplicación depende de la conexión a la red individual de cada usuario.

## Restricciones y técnicas de gestión

**Restricciones legales:** Es ilegal desarrollar una aplicación que acepte pagos a cambio de una moneda virtual con la intención de posteriormente retirar esta moneda virtual como dinero real.

**Restricciones técnicas:** Nuestra aplicación está destinada a usuarios Android aunque no descartamos una versión iOS y otra de escritorio para el futuro.

**Técnicas de gestión:** Hemos optado por un modelo descentralizado democrático en un comienzo, con el objetivo de evolucionar a un modelo descentralizado controlado tras un periodo de prueba de alrededor de tres semanas tras el que elegimos al responsable de organizar el grupo, pero siempre al mismo nivel jerárquico que los demás integrantes del grupo.

integrantes del grupo.

## 1.3 Modelo de proceso

Se ha decidido llevar a cabo el proyecto siguiendo la metodología Lean de desarrollo de software incorporando algún aspecto de otros métodos ágiles. como la figura del *product owner* del método scrum.

El motivo de adoptar esta metodología es el de realizar varias iteraciones con el objetivo de hacer un trabajo continuo que sea revisado con frecuencia y el de trabajar eficientemente como plantean los principios Lean de desarrollo de software:

1. Eliminar los desperdicios
2. Amplificar el aprendizaje
3. Decidir lo más tarde posible
4. Entregar tan rápido como sea posible
5. Capacitar al equipo
6. Construir integridad intrínseca
7. Visión de todo el conjunto

Es importante la figura del product owner en nuestra metodología, ya que sustituir al cliente por un integrante del equipo nos ayuda a especificar requisitos.