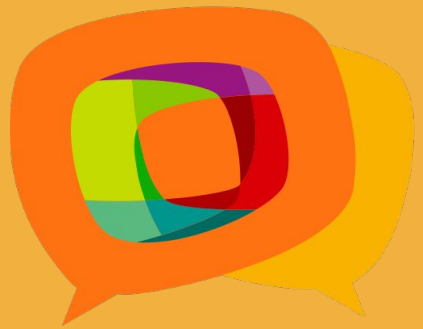
PWA

La resurrección del chat de Terra 🙊

Jorge Araneda

Vicente Calisto

Tristan Heuer



Hace ~20 años atrás…

chat
terra
terra

chat
terra
terra

Hoy

terra

Encuentre su noticia

CURIOSIDADES

Alonso “Monito” Vidal: 5 curiosidades del hijo mayor de Arturo Vidal que no conocías

Alonso cumplió 13 años el pasado 7 de mayo, y todos lo saludaron en sus redes.

SECRETOS

Esta es la mujer que rechazó a Chayanne pero aceptó a Luis Miguel

ENTRETENIMIENTO

INTIMIDADES

La verdad detrás del romance de la hija de José Luis 'el Puma' Rodríguez y Paul Walker

ENTRETENIMIENTO

¡Revivamos el **Chat de Terra!**

Ingresen a

https://pwa-grupo8.tk/

Frontend

Manifest

```
{
    "name": "Chat de Terra",
    "start_url": "/",
    "display": "standalone",
    "id": "grupo-8-pwa",
    "background_color": "#FFFFFF",
    "theme_color": "#ffa200",
    "icons": [
        {
            "src": "/favicon.png",
            "type": "image/png",
            "purpose": "maskable any",
            "sizes": "any"
        }
    ]
}
```

Service Worker: Registro

```
26  if ('serviceWorker' in navigator) {
27          navigator.serviceWorker
28            .register('/service-worker.js')
29            .then(function (registration) {
30              console.log(
31                'Registration successful, scope is:',
32                registration.scope
33              );
34            })
35            .catch(function (error) {
36              console.log('Service worker registration failed, error:', error);
37            });
38        }
```

Service Worker: Cache

```
21  self.addEventListener('install', async (event) => {
22    console.log('installing!');
23    const cache = await self.caches.open(CACHE_NAME);
24    await cache.addAll(urlsToCache);
25    self.skipWaiting();
26  });
```

```
28  self.addEventListener('activate', async (event) => {
29    console.log('activating!');
30    const keys = await caches.keys();
31    keys
32      .filter((key) => key !== CACHE_NAME)
33      .map((key) => event.waitUntil(caches.delete(key)));
34  });
35
```

```
36  async function networkFirst(request) {
37    const cache = await self.caches.open(CACHE_NAME);
38    try {
39      const response = await fetch(request);
40      cache.put(request, response.clone());
41      console.log('Network: ', request.url);
42      return response;
43    } catch {
44      console.log('Cache: ', request.url);
45      return cache.match(request);
46    }
47  }
48
49  self.addEventListener('fetch', (event) ⇒ {
50    if (event.request.method ≠ 'GET') return;
51    event.respondWith(networkFirst(event.request));
52  });
```

Token: Identificador de Instancia

```
getToken(messaging, { vapidKey })
        .then((currentToken) ⇒ {
          if (currentToken) {
            console.log(currentToken);
            sendRequest('/token', 'POST', { token: currentToken }).then(
              (response) ⇒ console.log('Send token response:', response)
            );
          } else {
            // Show permission request UI
            console.log(
              'No registration token available. Request permission to generate one.'
            );
          }
        })
        .catch((err) ⇒ {
          console.log('An error occurred while retrieving token. ', err);
        });
```

Notificaciones Push: Recepción

Firebase Cloud Messaging (FCM)

```
messaging.onBackgroundMessage((payload) => {
  console.log('Background: ', payload.data);

  console.log('Sending data to notification');
  try {
    self.clients.matchAll({ includeUncontrolled: true }).then((clients) => {
      clients.forEach((client) => {
        console.log('Sending to client ' + client);
        client.postMessage({
          data: payload.data,
        });
      });
    });
  } catch (e) {
    console.log(e);
  }

  // Customize notification here
  const notificationTitle = '¡Recibiste un nuevo mensaje!';
  const notificationOptions = {
    body: 'Ve al Chat de Terra para revisarlo',
    icon: '/favicon.png',
  };

  self.registration.showNotification(notificationTitle, notificationOptions);
});
```

Backend

Notificaciones Push: Envío

Firebase Cloud Messaging (FCM)

```go
func NewService(logger *log.Logger) (service, error) {
  opt := option.WithCredentialsJSON(config.FIREBASE)
  app, err := firebase.NewApp(context.Background(), nil, opt)
  if err != nil {
    return service{}, fmt.Errorf("error initializing app: %v", err)
  }

  msgClient, err := app.Messaging(context.Background())
  if err != nil {
    return service{}, fmt.Errorf("error initializing app: %v", err)
  }

  return service{app, msgClient, logger}, nil
}
```

```
45  func (s service) SubmitToken(ctx context.Context, token string) (*types.Error) {
46    resp, err := s.msgClient.SubscribeToTopic(ctx, []string{token}, "chat")
47
48    for _, e := range resp.Errors {
49      s.logger.Debug("%v", e.Reason)
50    }
51
52    s.logger.Info("Subscription: success: %v, failure: %v\n", resp.SuccessCount, resp.FailureCount)
53
54    if err ≠ nil {
55      s.logger.Critical(err)
56      return types.UnexpectedError
57    }
58
59    return nil
60  }
```

```go
func (s service) SendMessage(ctx context.Context, user string, content string) (*types.Error) {
  message := &messaging.Message{
    Data: map[string]string{
      "user": user,
      "content": content,
    },
    Topic: "chat",
  }
  // Send a message to the devices subscribed to the provided topic.
  response, err := s.msgClient.Send(ctx, message)

  s.logger.Debug("SEND: %v\n", response)
  if err ≠ nil {
    s.logger.Critical(err)
    return  types.UnexpectedError
  }
  return nil
}
```