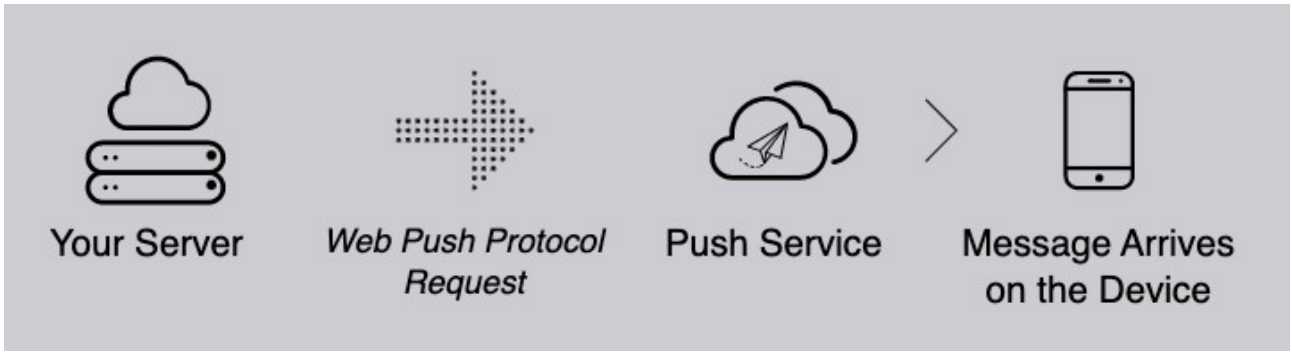
How do push notifications work?

1. Adding **client** logic to ask the user for permission to send push notifications. Once we have permission a **PushSubscription object** is return to us by the browser (which can be thought of as a device/client identifier). We then send this object to our server (to store in database)

```
// Use serviceWorker.ready to ensure that you can subscribe for push
  navigator.serviceWorker.ready.then((registration) => {
    const options = {
      userVisibleOnly: true,
      applicationServerKey: publicVapidKey
    }
    registration.pushManager
      .subscribe(options)
      .then((pushSubscription) => sendSubscriptionObjectToServer(pushSubscription))
      .catch((err) => console.log('There was an error ', err))
  })
```

```
{
  endpoint: 'https://fcm.googleapis.com/fcm/send/dnllPq6NR7E:APA91bERryUT-d2C',
  expirationTime: null,
  keys: {
    p256dh: 'BK6Yl1dbZTCCfVKeKTPDsUajF-kMc4eGaewNzvFspZPkoX2F__HHuSj1buJNg0Zq9JF8lTYkfHZH2lv8UDBqqLI',
    auth: '_WaWEIbW1xnIE4LSaho9uw'
  }
}
```

2. Once we receive the pushSubscription object(s) from various clients. We add **server** logic to push messages to client devices (via browser vendor push service) – the pushSubscription object(s) let's the Push Service know which client to send the push to.



3. We then add **client** logic to receive messages that have been pushed to the device and display them as notifications (using service worker)

```
1  //Listen for Push event from our Web Server
2  self.addEventListener('push', (event) => {
3    console.log('Recieved push notification from Push Service')
4    //Notifications sent from server e.g 50% off sale!!!
5    const data = event.data.json()
6    self.registration.showNotification(data.title, {
7      body: data.message,
8      icon: 'http://image.ibb.co/frYOFd/tmlogo.png'
9    })
10 })
11
```