request poll()

Name

request-poll - Wait for some event on a file descriptor

Synopsis

```
#include <sys/poll.h>
```

Arguments

ufds
: List of file descriptor events to be watched

nfds
: Number of file descriptor events at the *ufds array

timeout
: Timeout to wait for events

Description

With the :c:func:`poll()` function applications can wait for a request to complete.

On success :c:func:`poll()` returns the number of file descriptors that have been selected (that is, file descriptors for which the `revents` field of the respective struct :c:type:`pollfd` is non-zero). Request file descriptor set the `POLLPRI` flag in `revents` when the request was completed. When the function times out it returns a value of zero, on failure it returns -1 and the `errno` variable is set appropriately.

Attempting to poll for a request that is not yet queued will set the `POLLERR` flag in `revents`.

Return Value

On success, :c:func:`poll()` returns the number of structures which have non-zero `revents` fields, or zero if the call timed out. On error -1 is returned, and the `errno` variable is set appropriately:

> **System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\mediactl\[linux-master][Documentation][userspace-api][media][mediactl]request-func-poll.rst`, line 56); _backlink_**
>
> Unknown interpreted text role "c:func".

`EBADF`
> One or more of the `ufds` members specify an invalid file descriptor.

`EFAULT`
> `ufds` references an inaccessible memory area.

`EINTR`
> The call was interrupted by a signal.

`EINVAL`
> The `nfds` value exceeds the `RLIMIT_NOFILE` value. Use `getrlimit()` to obtain this value.