

## Regression Test Report

Kenneth Link

### Description:

This report is being created to document exploratory testing on the 393torrent application. The test procedure was completed on an amazon ec2 machine running python 2.7.6.

### Procedure:

1. Attempted to use a torrent file that doesn't exist
  - a. Verified the error message appeared

```
root@domU-12-31-39-0A-16-99:/home/ubuntu/393torrent# python torrent/torrent.py Tiny\ Core\ Linux\ Core\ 5.1\ \ (1\).torrent
Sorry! File doesn't appear to be readable!
Error: [Errno 2] No such file or directory: 'Tiny Core Linux Core 5.1 (1).torrent'
```

2. Attempted to use a broken torrent file
  - a. Verified an error message appeared

```
root@domU-12-31-39-0A-16-99:/home/ubuntu/393torrent# python torrent/torrent.py README.md
Traceback (most recent call last):
  File "torrent/torrent.py", line 165, in <module>
    TorrentApplication(args.torrent_file, args.seed).start()
  File "torrent/torrent.py", line 19, in __init__
    self.file_info = TorrentFileRetriever(torrent_file).retrieve()
  File "torrent/torrent.py", line 35, in retrieve
    return self.handle_torrent_file(file_text)
  File "torrent/torrent.py", line 41, in handle_torrent_file
    return fileinfo.TorrentFileInfo(torrent_file_text)
  File "/home/ubuntu/393torrent/torrent/fileinfo.py", line 12, in __init__
    torrent_dict = bencode.bdecode(torrent_file_text)
  File "/usr/local/lib/python2.7/dist-packages/bencode/__init__.py", line 68, in bdecode
    raise BTFailure("not a valid bencoded string")
bencode.BTL.BTFailure: not a valid bencoded string
```

3. Attempted to use a torrent file in which the tracker is unresponsive
  - a. Verified an error message appeared

### Results:

Steps 1 and 3 verified successfully, however a broken torrent causes a stacktrace.

### Conclusions:

An error message should be created for broken torrent files.