P2P FILE SHARING

‘The Octopus’ Group  
  
Team Members:

AHMAD, FARHAN

ALIPOURSIMAKANI, KAMRAN

ANDERSSON EKSTRÖM, MAX

BERNTSSON, FREDRIK

CHADALAPAKA, GAYATRI

IQBAL, NAYYAR

KUKKAPALLI, NAGA VYSHNAVI

NYHLÉN, JESPER

ROUTHU, VENKATA SAI KALYAN

SHAD MANFEAT, SEYEDEH MERSEDEH

ZAREI, KAMBIZ

Type of Document: Acceptance test plan

Version 1.0

Publication Date: May 8th ,2016

**1. PREFACE:**

***Section 2*: Glossary and abbreviations**

**Section 3: Acceptance Test Plan**

**Section 4: References**

**Release v1.1 on 2016-05-15**

-updated to only show user requirements

**Release v1.0 on 2016-05-08**

- Initial release

**2: Glossary and abbreviations**

Dark Peer - Is the peer which exceeds the valid time interval assigned by the bootstrap server and thus removed from list of known peers, it will be marked and known as Dark Peer in the network.

Dark Content - Is the files located on any dark peers.

Swarm - The main file which each peer gets from the server at its first connection which includes list of shared files, list of peers and information about which node shares the file and the swarm metadata.

Swarm Metadata - Includes filenames and file message digest

File Metadata - Is the set of headers combined with the filename.

Bootstrap-Server - Server which will inform the other nodes about the presence of any node connected to network and will fetch the swarm metadata from them.

Service: simply is the outcome of chain of functions to reach it designed and defined purpose.

Server: the device which is going to provide the services by using its resources.

Interface: is the presentation of results which is done by the servers based on service, and it is the visible part of a system architecture.

**3: Acceptance Test Plan**

The user requirements to be fulfilled by our system will be listed below with at least one test case to show that the system works as intended.

For full information:

-on the requirements, check the provided requirement document [1].

-on the test cases, check the provided test document [2].

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement ID** | **Description** | **Test Case ID** | **Description** |
| Req-Front\_130 | The peers shall provide an option to download a swarm in the GUI. | Test\_102 | To test whether it´s possible to download a swarm |
| Req-Front\_131 | The peers shall provide an option to create swarms in the GUI | Test\_101 | To test whether it´s possible to create swarms in the GUI. |
| Req-Front\_132 | The peers shall provide an option to see a list of files in the GUI | Test\_101 | To test whether it´s possible to create swarms in the GUI. |
| Req-Front\_133 | The peers shall provide an option to search for files in the GUI | Test\_103 | To test if you can search for files in the GUI |
| Req-Front\_135 | The peers shall provide an option to delete a swarm in the GUI. | Test\_105 | Testing to delete a swarm from the GUI |
| Req-Front\_136 | The peers shall provide an option to choose whether a swarm is dark or visible in the GUI upon creation. | Test\_106 | Testing if you can set a swarm as dark in the GUI upon creation |
| Req-Front\_137 | The peers shall show the progress of downloads. | Test\_108 | Testing if you can see progress of downloads |
| Req-Front\_138 | The peers shall show the estimated time to complete downloads | Test\_109 | Test if you can see an estimate of how long is left of downloads |
| Req-Front\_139 | The peers shall show the list of the IP addresses associated with a swarm | Test\_110 | Test if you can see all IP addresses which the swarm can be downloaded from |
| Req-Front\_140 | The peers shall show all IP addresses of connected peers. | Test\_111 | Test if you can see IP addresses of the connected peers |
| Req-Front\_141 | The peers shall show an error message in case a file digest fails and provide an option to redownload the file. | Test\_112 | Test if you can redownload a file in case a file digest fails |

**4. REFERENCES:**

[1] “The Octopus” group, 2015. Test Document v1.2.

[2] “The Octopus” group, 2015. Requirements Document v1.3.