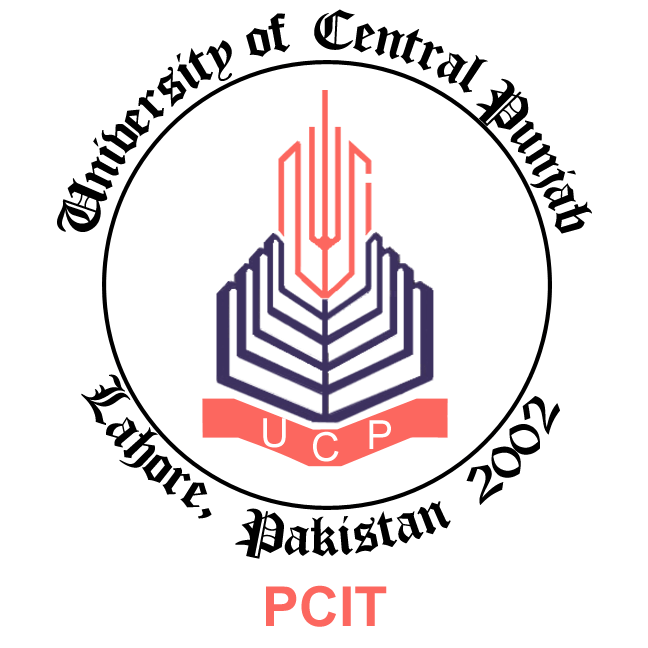
**BSCS FINAL PROJECT PROPOSAL**

Magneto (A Torrent Client)

*Term of Registration: Fall 2014*



Presented by:

|  |  |
| --- | --- |
| **Registration No:** | **Name:** |
| L1F20BSCS0603 | Ali Anser |
| L1S21BSCS0065 | Minhaal Ali Hafeez |
| L1S23BSCS0257 | Faheem Raza |

|  |
| --- |
| Faculty of Information Technology |

University of Central Punjab

**Project Title**

Mangeto (A Torrent Client)

**Project Advisor**

Ghulam Mustafa

**Particulars of the students:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Registration#**  eg.**L1F00BSCS0101** | **Name in Full**  Use Block Letters | **CGPA** | **Signatures** |
| 1 | L1F20BSCS0603 | ALI ANSER | 2.82 |  |
| 2 | L1S21BSCS0065 | MINHAAL ALI HAFEEZ | 2.41 |  |
| 3 | L1S23BSCS0257 | FAHEEM RAZA | 2.88 |  |

**Advisor’s Consent**

I Prof./Dr./Mr./Ms. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ am willing to guide these students in all phases of above-mentioned project as advisor. I have carefully seen the Title and description of the project and believe that it is of an appropriate difficulty level for the number of students named above.

|  |  |  |
| --- | --- | --- |
| **Note:**  Advisor can’t be changed without prior permission of the Manager Projects and the duration for completion of the Project is 2 regular semesters (approx.) from the date of Registration of Research Project. | Signatures and Date  |  | | --- | |  |   **Advisor** |

**EVALUATOR/REFEREE 1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| I have carefully read the project proposal and feel that the proposed project is a useful one and of a sufficient difficulty level to justify 2 regular semesters workload for above mentioned students. I have made recommendations in the evaluation form to improve the scope and quality of the project. | | | | | |
|  | | | | Signatures and Date | |
|  |  |  |  |  |  |
|  | | | |  |

**EVALUATOR/REFEREE 2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| I have carefully read the project proposal and feel that the proposed project is a useful one and of a sufficient difficulty level to justify 2 regular semesters workload for above mentioned students. I have made recommendations in the evaluation form to improve the scope and quality of the project. | | | | | |
|  | | | | Signatures and Date | |
|  |  |  |  |  |  |
|  | | | |  |

**Abstract / Executive Summary**

Magneto will be a cross-platform torrent client that integrates a built-in VPN module. The primary goal is to provide a secure, private, and seamless torrenting experience by eliminating the need for external VPN services. Magneto aims to enhance user privacy, reduce configuration complexity, and deliver a reliable peer-to-peer file-sharing environment.

Some questions arise such as “Why not just add a VPN module to existing torrent clients?” or “Why not use existing torrenting libraries with VPN libraries and just integrate them?”. We did research on this and found out that it is not feasible to modify existing torrenting clients or use a torrenting library along with VPN library to create Magneto.  
  
Existing torrent clients’ and libraries’ source code is not written in such a way to support such integration. We would have to modify thousands of lines of code related to the BitTorrent protocol and networking component to get remotely close to our goal, which is not feasible.

So, we will be implementing everything from scratch.

We predict that there will be market value for Magneto among avid torrent users. Using VPNs while torrenting is a rather popular idea but there is no such client that has a builtin VPN module, which is why we are proposing Magneto.

**Introduction and Background**

Torrenting is a popular method for sharing large files through peer-to-peer (P2P) communication. Despite its efficiency, it poses significant privacy and security concerns, as user IP addresses are exposed to other peers in the network. This exposure can make users vulnerable to surveillance, tracking by ISPs and malicious attacks on their machine.

VPNs help by encrypting internet traffic and masking user IP addresses, providing anonymity and enhanced security. Traditional torrent clients require users to separately configure and use VPN services to safeguard their online identity, which can be cumbersome. This project seeks to solve that problem by embedding a VPN directly within the Magneto torrent client.

**Statement of the Problem**

Existing torrent clients lack integrated privacy mechanisms, requiring users to rely on external VPNs or proxies. This setup can be cumbersome and prone to errors, discouraging users from protecting their anonymity. Additionally, dependence on third-party tools may compromise usability and reliability. Magneto addresses these issues by providing a seamless, privacy-focused torrenting solution.

**Objective(s) / Aim(s) / Target(s)**

* Develop a cross-platform torrent client that adheres to the BitTorrent protocol while ensuring robust functionality.
* Integrate a VPN module to protect user privacy and maintain anonymity during torrenting, mitigating risks such as surveillance and ISP throttling.
* Deliver a user-friendly interface to simplify torrent management and increase accessibility.
* Optimise performance to efficiently handle multiple simultaneous downloads and uploads without sacrificing speed or reliability.
* Simplify configuration by offering seamless, built-in privacy features, eliminating dependence on third-party tools.

**Completeness Criteria**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Criteria** | **Weightage %** |
| 1 | Desktop GUI | 10 |
| 2 | **VPN Integration and Security Features** | 20 |
| 3 | **Core BitTorrent Protocol Implementation** | 40 |
| 4 | **Torrent File and Magnet URL Parsers** | 15 |
| 5 | Documentation | 5 |
| 6 | Performance | 10 |

**Desktop GUI (10%)**

The Desktop GUI criterion evaluates the user interface and user experience of the Magneto torrent client. The GUI should provide a well-organized, easy-to-navigate design that allows users to interact efficiently with the application. It should support key actions such as initiating and managing torrent downloads, configuring VPN settings, and displaying important information like download progress, file sizes, and peers. The interface should be intuitive, responsive, and adaptable across different screen sizes and platforms, ensuring a smooth experience for users.

**VPN Integration and Security Features (20%)**

This criterion focuses on the integration of a VPN module into the torrent client to provide secure, anonymous downloading. The VPN should ensure that user traffic is encrypted and that no IP address leaks occur during file transfers. The client should support the configuration of VPN settings, initiate or terminate the VPN connection, and handle automatic reconnection. Security features also include ensuring the safety of data through encryption protocols and preventing any potential breaches or privacy leaks during torrenting.

**Core BitTorrent Protocol Implementation (40%)**

The core BitTorrent protocol implementation is the heart of the client. It involves enabling peer-to-peer communication, ensuring efficient downloading and uploading of torrent files, and handling multiple connections concurrently. The client should correctly download pieces of files from various peers and verify their integrity using hashes or checksums. Torrent metadata (from .torrent files) should be accurately parsed, and the application should be able to join a swarm and manage file transfers effectively. This criterion also includes supporting the seeding of downloaded files and maintaining proper peer connections.

**Torrent File and Magnet URL Parsers (15%)**

This criterion assesses the functionality of the torrent file and Magnet URL parsers. The bencode format parser should be implemented to handle the extraction of necessary metadata from .torrent files, such as file names, tracker information, and piece data. Additionally, the client should support Magnet URLs, which require a parser to extract relevant information like the hash and associated trackers. Both parsers should work accurately and efficiently, handling different types of torrent files and Magnet links without errors.

**Documentation (5%)**

Comprehensive documentation is essential for both end-users and developers. This includes user documentation that provides clear instructions on how to install, configure, and use the torrent client, as well as troubleshooting guidance. Technical documentation should explain the system architecture, algorithms used, and code structure, allowing future developers to understand and extend the project easily. Code comments should be present throughout the codebase, ensuring that key functionalities and design decisions are well-explained for readability and maintainability.

**Performance (10%)**

The performance criterion ensures that the client is efficient in terms of speed and resource usage. The client should be capable of achieving fast download speeds by making optimal use of available bandwidth and handling multiple peers effectively. The application should not consume excessive CPU or memory resources, especially during large downloads or when handling a high number of peers. The client should also scale well, maintaining stability and responsiveness even when managing large torrents or when the number of peers increases.

**Challenges**

* **VPN Integration:** Implementing a robust and secure VPN solution compatible with P2P traffic.
* **BitTorrent Specification:** Understanding BitTorrent specification to the core.
* **Performance Optimization:** Ensuring that VPN usage does not degrade torrent download/upload speeds.
* **Cross-Platform Compatibility:** Maintaining consistent performance and interface across different operating systems.
* **Security Measures:** Protecting user data and preventing IP leakage.
* **Protocol Compliance:** Adhering to the BitTorrent specification while adding VPN functionalities.

**Knowledge Areas Required**

* Networking and VPN technologies.
* Peer-to-peer communication protocols, specifically BitTorrent.
* Cross-platform application development using frameworks like Qt.
* Software development best practices and version control with Git.
* Cryptography and secure communication techniques.

**Learning Outcomes**

* Mastery of networking concepts and VPN integration.
* In-depth understanding of P2P communication and the BitTorrent protocol.
* Proficiency in cross-platform software development.
* Problem-solving skills for optimising secure and efficient torrenting solutions.
* Experience with project planning, development, and testing.

**Nature of the End Product / Research Outcomes**

Magneto will be a fully functional torrent client with a built-in VPN module. It will provide users with secure and anonymous torrenting capabilities, a user-friendly interface, and robust performance. Additionally, the project will contribute to research on the integration of privacy-enhancing technologies within P2P applications.

**Related Work / Literature Survey / Literature Review**

Numerous torrent clients, such as uTorrent, qBittorrent, and Transmission, are known for their efficient and user-friendly interfaces. These clients follow the BitTorrent protocol but lack built-in privacy features. Users often resort to third-party VPN services to maintain anonymity. While this approach offers some protection, it complicates the user experience.

Research on privacy-preserving techniques in P2P networks highlights the importance of anonymity and secure communication. Studies have demonstrated that integrating privacy features directly within applications can improve adoption rates and usability. By building on these findings, Magneto aims to set a precedent for privacy-focused torrent clients.

**Deliverables / Work Breakdown Structure**

**Research Phase:**

* Study BitTorrent specification and VPN technologies.
* Review existing torrent clients and privacy solutions.

**Development Phase:**

* Implement baseline GUI using Qt.
* Develop a file manager for download paths and metadata.
* Implement peer-to-peer communication.
* Integrate VPN module.

**Testing Phase:**

* Perform unit and integration testing.
* Evaluate performance with and without VPN.
* Conduct security assessments to prevent IP leakage.

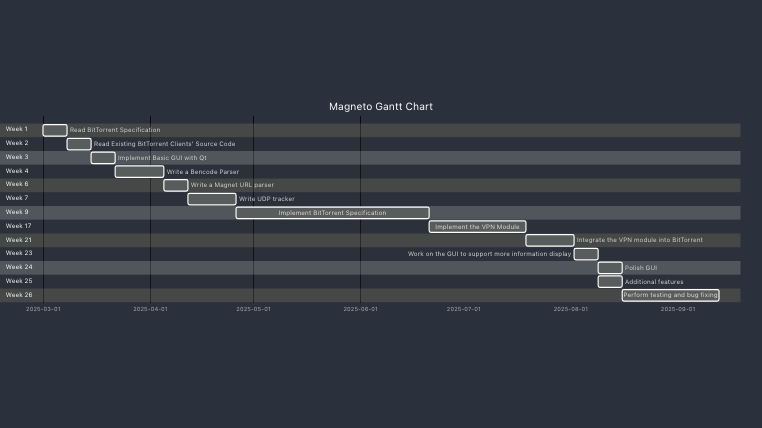
**Deployment Phase:**

* Develop a website for client downloads and source code access.
* Provide user documentation and support.

**Final Deliverables:**

* Fully functional cross-platform torrent client with VPN integration.
* Technical documentation and user manual.
* Website for client distribution.

**Project Plan / Project Schedule / Project Timetable / Project Calendar**

  
We will work on Magneto based on the above Gantt chart and we intend to touch base with the advisor every week.

If a new feature is supposed to be developed that week, we will get acceptance of our advisor for that feature.

If it’s a continuation week of an existing feature then we will show progress of that feature to our advisor get his approval.

**Resources Required**

We do not require any resources from the university for development of this project.

**Miscellaneous**

**Collaboration Tools:** Utilize Git for version control to manage project code and documentation collaboratively.

**Time Management:** Follow a feature-driven development approach with well-defined milestones for each project phase to ensure timely completion.

**Testing Environment:** Set up isolated testing environments to evaluate VPN performance and functionality without compromising real network security.

**Documentation:** Maintain comprehensive technical and user documentation throughout the project.

**Security Best Practices:** Regularly audit the code for security vulnerabilities, particularly in VPN integration and P2P communication.

**Abstract Story Board and Identification of Characters (For Game-Oriented Projects Only)**

N/A.

**Sketch of Proposed Solution (For Research-based and Hardware-Oriented Projects Only)**

N/A.

**References/Bibliography**

* <https://wiki.theory.org/BitTorrentSpecification>
* <https://nordvpn.com/blog/interesting-vpn-uses/>
* <https://support.torproject.org/misc/misc-4/>
* Research on VPN and Privacy in Peer-to-Peer Networks, Journal of Secure Computing, 2023.
* uTorrent and qBittorrent official documentation for feature comparisons.
* Qt Framework Documentation: [https://doc.qt.io](https://doc.qt.io/)