**SIT725 – APPLIED SOFTWARE ENGINEERING**

**TASK 4.1P - Software Engineering Ethics**

* **Ethical Principles**

1. User Privacy and Consent

* Players should be required to give explicit consent before the data is accessed
* Deliver easily understandable information concerning the data collection and usage
* Users can opt-out or turn off location tracking whenever they want

1. Data Protection and Security
   * Adopt strong encryption for user location and amusement details
   * Gather the minimal amount of data that is strictly necessary for the core functionalities of the application
   * Actively guard user data from unauthorized access or leaks
2. Transparency and Accountability
   * Show how the location-based function operates and its restrictions in an unmistakable way
   * Enable users to regulate the amount of information they share explicitly
   * Display a readable privacy policy and terms of service
3. Non-Discrimination
   * Check if the application treats all users, no matter the class
   * Avoid biases in recommendations and accessibility elements of locations
   * Use reliable user interfaces and experiences
4. Ethical Use of Technology
   * Curb the likelihood of any potential abuses of location tracking
   * Establish measures to prevent stalking or harassment
   * Observe individual privacy and personal boundaries

* **Software Quality Characteristics**

1. Reliability
   * Accurate and consistent socket location information
   * Introduce error handling and graceful failure strategies
   * Ensure interoperable performance across different devices and networks
2. Usability
   * Design an interface that is easy to understand and intuitive
   * Create routes and socket location instructions that are very clear
   * Give useful feedback and instructions to the users
3. Efficiency
   * Have a careful look at locating algorithms for improvements
   * Lowest battery and resource consumption are ensured for this purpose
   * Instant and immediate socket location results
4. Maintainability
   * Use only neat, modular codes and label them with further comments
   * Design for adaptability and include the groundwork for future feature expansion
   * Adhere to the coding practices along with software architecture techniques prescribed
5. Security
   * Implement reliable authentication schemes
   * Defend against the possible security holes
   * Frequently upgrade and patch the software
6. Compatibility
   * Make sure that the app can work across all platforms
   * Allow devices running on different operating systems to work together
   * Create customizable apps to cater to the customer's needs on various platforms
7. Scalability
   * Develop the application to allow for the addition of more users
   * Show an architecture that can grow with your future additions
   * Minimize database and server-sided operations