**10. Tétel**

10. Egy táblajáték program tesztelését kell elvégezni. Feladata, hogy a béta állapotú szoftverben hibákat tárjon fel.

* Tegyen javaslatot egy teljes körű tesztelési tervre!
* Határozza meg a „jó teszteset” kritériumait!
* Tegyen javaslatot a tesztelési dokumentum formájára és felépítésére!
* Ismertesse egy hibakezelő rendszer működését!

**Tesztelési terv**

A táblajáték program tesztelésének teljes körűnek kell lennie, hogy biztosítsa a játék megfelelő működését és hibamentességét. A tesztelési tervet funkcionális, teljesítmény, kompatibilitási, biztonsági, felhasználói és esettesztelések, valamint integrációs tesztelések bevonásával kell elvégezni. Így a játék alapfunkcióit, teljesítményét, kompatibilitását, biztonságát, felhasználói élményét, esetekre való alkalmazhatóságát és moduljainak integrációját tesztelik, hogy biztosítsák a játék biztonságos, megbízható és élvezhető voltát a felhasználók számára.

A teljes körű tesztelési terv a következő lépéseket tartalmazhatja:

1. **Funkcionális tesztelés:** Ellenőrizze a játék alapfunkcióit, mint például a játékmódok, a játékmenet, a pontszámítás, a játékosok közti kommunikáció és a beállítások.
2. **Teljesítménytesztelés:** Ellenőrizze a játék teljesítményét, például a betöltési időt, a frissítési sebességet és a memóriahasználatot.
3. **Kompatibilitási tesztelés:** Ellenőrizze, hogy a játék kompatibilis-e az eltérő eszközökkel, operációs rendszerekkel és böngészőkkel.
4. **Biztonsági tesztelés:** Ellenőrizze a játék biztonsági funkcióit, például a jelszóvédelmet és az adatvédelmet.
5. **Felhasználói tesztelés:** Ellenőrizze a felhasználói élményt, például a felhasználói felület, a navigáció és a játék egyéb aspektusainak egyszerűségét.
6. **Esettesztelés:** Ellenőrizze a játékot különféle esetekben, például a váratlan események vagy hibák esetén.
7. **Integrációs tesztelés:** Ellenőrizze a játék különböző moduljainak integrációját, hogy biztosítsa, hogy az egyes részek megfelelően működjenek együtt.

**A jó teszteset kritériuma**

A jó tesztesetnek több kritériumnak kell megfelelnie, hogy hatékony és megbízható tesztelést biztosítson. Az egyik legfontosabb kritérium, hogy a teszteseteknek tisztán és egyértelműen definiáltnak kell lenniük, hogy mindenki, aki használja őket, ugyanazt az eredményt kapja. A teszteseteknek a játék funkcióinak és követelményeinek teljes körét lefedi. Kiterjednek a rendszer normál működésére, valamint a rendszer hibás működési állapotaira is. A teszteseteknek áttekinthetőnek és megismételhetőnek kell lenniük, és egymástól eltérő eredményeket kell adniuk, amennyiben a játék állapota más. Végül a teszteseteknek egyszerűen automatizálhatónak kell lenniük, hogy a tesztelési folyamat hatékonyabb és gyorsabb legyen.

**Tesztelési dokumentum**

A tesztelési dokumentum egy olyan dokumentum, amely részletesen dokumentálja a szoftver tesztelésének folyamatát, a tesztelési eseteket, az eredményeket és az esetleges hibákat. Az ilyen dokumentumok segítenek nyomon követni a tesztelési folyamatot, megerősíteni a teszteredményeket és javítási ajánlásokat adni a fejlesztőknek a hibák kijavításához. A tesztelési dokumentumok fontos részei a szoftverfejlesztési folyamatnak, mert segítenek biztosítani a szoftver minőségét és megfelelőségét.

A táblajáték tesztelési dokumentumának felépítése először is tartalmaznia kell a tesztelés céljait és azokat a funkciókat, amelyeket tesztelni szeretne. Azután a tesztek lépéseit kell részletesen leírni, beleértve a teszteseteket, az eseteket és az esetek várható eredményeit. Az egyes tesztek elvégzéséhez szükséges eszközöket és eszközöket is le kell írni. A dokumentáció továbbá tartalmazza a teszteredmények dokumentálására szolgáló folyamatot, beleértve a hibák rögzítését, a hibák követését és a hibák javítását. Az eredmények összesítése és elemzése is szerepelnie kell a dokumentumban. Végül, a dokumentáció tartalmaznia kell a tesztelés befejezését követő lépéseket, például a javításokat és a dokumentáció frissítését. Az egész tesztelési dokumentáció jól szerkesztett, logikai, részletes és könnyen olvasható formában kell lennie.

**Hibakezelő rendszer**

A hibakezelő rendszer a szoftverfejlesztés során a rendszer hibáinak, problémáinak és kérdéseinek nyomon követésére és megoldására szolgáló eszköz. A hibakezelő rendszer lehetővé teszi a fejlesztők számára, hogy nyomon kövessék a hibákat, amikor a rendszer használatban van, és megoldást javasoljanak rájuk. A hibakezelő rendszer jellemzően tartalmaz egy hibajegy nyomkövetési rendszert, amelyen keresztül a fejlesztők nyomon követhetik a hibák állapotát, illetve egy eszközöket, hogy megoldást javasoljanak rájuk. A hibakezelő rendszer kulcsfontosságú a szoftver minőségének javítása és a fejlesztők munkájának hatékonyságának növelése érdekében.

**Visual Studio Team Services**

A Visual Studio Team Services (VSTS) hibakezelő rendszere egy eszköz, amely segít a fejlesztőknek a hibák könnyebb azonosításában és kijavításában a fejlesztési folyamat során. A VSTS hibakezelő rendszere integrálva van a Visual Studio fejlesztői eszközökbe, így a fejlesztők automatikusan nyomon követhetik a hibáikat, amint azokat javítani kezdenek. A rendszer támogatja a hibák kategorizálását, prioritásukat, státuszukat és egyéb információikat, hogy könnyebb legyen nyomon követni őket. A VSTS hibakezelő rendszerének további előnyei közé tartozik a hibák megosztása, a kommunikációs eszközök könnyű hozzáférése, valamint az összes fejlesztői tevékenység egységes nyomon követése a fejlesztési folyamat során.