

České vysoké učení technické

Fakulta elektrotechnická

PLÁN TESTŮ

Systém elektronických voleb

E-volby

OBSAH

Obsah	2
1. Úvod	3
2. Testování jednotek - White-box testing	3
2.1 Definice White-box testování	3
2.2 Použití v projektu	3
3. Integrační testování.....	3
3.1 Přístup k integračnímu testování	3
3.2 Integrační testování projektu.....	3
4. Validační testování - Black-box testing.....	4
4.1 Definice Black-box testování.....	4
4.2 Použití v projektu	4
4.3 Testování webového rozhraní.....	4
5. Systémové testování.....	4
6. Výsledky testů.....	5

1. ÚVOD

Účelem testování je odhalení chyb vzniklých při programování, chování aplikace v nestandardních situacích a zajištění bezproblémového chodu aplikace v jednotlivých uživatelských scénářích. V plánu testování projektu e-volby je využíváno především *White-box* a *Black-box* testování.

2. TESTOVÁNÍ JEDNOTEK - *WHITE-BOX TESTING*

2.1 Definice *White-box* testování

White-box testování používá informace o vnitřní struktuře testovaného modulu a znalosti zdrojového kódu.

2.2 Použití v projektu

White-box testování v projektu e-volby zajišťují jednotkové testy, konkrétně je to testovací framework JUnit, integrovaný do vývojového prostředí NetBeans.

Jednotkové testy fungují na principu rozdělení aplikace na malé části a testování těchto částí bez závislosti na ostatních.

3. INTEGRAČNÍ TESTOVÁNÍ

3.1 Přístup k integračnímu testování

V projektu E-volby je pro integrační testování použit přístup "velkého třesku". Nejdříve se spojí všechny moduly a pak je testováno vše naráz. Projekt není rozsáhlý, proto je možné tento přístup použít.

3.2 Integrační testování projektu

Integrační testování se zaměřuje na vlastní přístup k datům a komunikaci. V případě projektu E-volby jsou to EJB (*Enterprise JavaBean*) moduly. Automatické testování EJB modulů je v projektu e-volby obtížné protože na automatické testování nebylo myšleno při

zahájení vývoje. Testování je tedy zajištění manuálně, přímou kontrolou obsahu tabulek v databázi po provedení uživatelských akcí. Tato metoda je jednoduchá, ale časově náročná. Je to však efektivnější řešení než přepracování projektu pro umožnění automatického testování a poté samotné psaní automatických testů.

4. VALIDAČNÍ TESTOVÁNÍ - *BLACK-BOX TESTING*

4.1 Definice *Black-box* testování

Black-box testování uvažuje aplikaci jako černou skříňku, bez znalosti vnitřní struktury a zdrojového kódu. Na rozdíl od White-box testování tedy ověřuje vstupy a výstupy aplikace místo fungování jednotlivých částí zdrojového kódu.

4.2 Použití v projektu

Black-box testování je v projektu e-volby použito jako testování celkové funkčnosti projektu, tedy funkčnosti webové uživatelského rozhraní.

4.3 Testování webového rozhraní

Testování webového rozhraní je zajištěno pomocí testovacího frameworku Selenium. Tento framework umožňuje nahrát skripty standardních uživatelských akcí, jejich automatické či poloautomatické provedení a kontrolu správnosti provedení. Testerovi tedy skript umožňuje rychlou a spolehlivou kontrolu, zda po změně zdrojového kódu aplikace stále funguje z uživatelského pohledu tak, jak by měla.

5. SYSTÉMOVÉ TESTOVÁNÍ

Systémové testování tvoří hlavní část vývojového testování a ověřuje celkovou funkčnost aplikace. Je zde testováno především splnění požadavků na software a chování v nestandardních situacích.

Zátěžový test lze provést monitorováním systémových prostředků (zatížení procesoru, paměti, síťové komunikace a potřeba místa na disku) při běžném provozu a při simulování nestandardních situací.

Testování obnovy se týká uložení dat. Je třeba otestovat plánované zálohy dat z databáze, funkčnost vrácení databáze do stavu před chybou a schopnost obnovit data po restartu či pádu databáze a aplikace. Testování obnovy se bude týkat reálně nasazeného databázového stroje a práce s ním.

Bezpečnostní testování musí prověřit zabezpečení systému před útoky zvenčí a manipulaci s daty. Projekt elektronických voleb je záležitost, která musí být naprosto bezpečná a je třeba zaručit, že data udržovaná v aplikaci nemohou být upravována. Bezpečnostní testování může být provedeno zkouškou získání neautorizovaného přístupu k datům přes samotnou aplikaci nebo přímo fyzický stroj

6. VÝSLEDKY TESTŮ

Výstupy z testování budou obsahovat informace o provedení, obsahu a výstupu daného testu. V případě neúspěšného testu musí výsledek obsahovat důvod neúspěchu a možné kroky k odstranění problému s neúspěšným testem.