# Úvod:

Testovat budeme web, sloužící k vytváření poznámek. Na web je třeba se přihlásit a není-li uživatel přihlášen, musí se zaregistrovat. Uživatel může poznámky přidat, označit za hotové či smazat. Každá poznámka obsahuje název, datum do kdy musí být dokončena, stupeň důležitosti a nakonec samotný popis. Pomocí filtru si uživatel může vyfiltrovat poznámky podle důležitosti. V nastavení si uživatel může nastavit vzhled stránky. Každý uživatel má po úspěšné registraci a přihlášení přístup pouze ke svým vlastním poznámkám.

# Popis:

K realizaci byly využity technologie PHP, Javascript, HTML5 a CSS.

HTML5 a CSS byly použity pro vytvoření struktury a stylu stránky

Zbylé dvě byly použity na back end, přesněji obsluha formulářů, přidávání poznámek nebo přidávání dat do databází

## Bezpečnost

Na stránce bylo použito několik opatření proti útokům typu XSS a CSRF.

### htmlspecialchars()

Jedním bylo použití funkce htmlspecialchars() na všechna data získaná z formulářů. Tato funkce zabraňuje útokům XSS tím, že konvertuje speciální znaky na jejich HTML entity, což znemožňuje útočníkovi vložit škodlivý kód do stránky.

### unikátní token

Při každém odeslání formuláře na vytvoření poznámky je vygenerován unikátní token, který je uložen v session uživatele a také je vložen do formuláře jako skrytý input. Při odeslání formuláře ověřujeme, zda token odpovídá tokenu uloženému v session.

## Databáze

Databáze je vytvořena ve formátu jSON. Byly do ni přidávány pole s konkrétními daty pomocí funkce json\_decode a json\_encode a skládá se přesně ze dvou souborů

### users.json

Každý nový uživatel se přidá do souboru users.json pouze v případě, validní registrace. Do databáze se uloží následující informace:

* jeho unikátní ID
* uživatelské jméno
* email
* zahashované heslo
* náhodně vygenerovaná sůl
* téma (defaultně nastaveno na světlé)

### notes.json

Při úspěšném přihlášení a přidání validní poznámky, se přidá do souboru notes.json následující informace:

* unikátní ID poznámky
* její název
* datum do kdy má být hotova
* důležitost
* samotný text poznámky
* datum vytvoření
* ID uživatele, který ji přidal

# Testovací strategie:

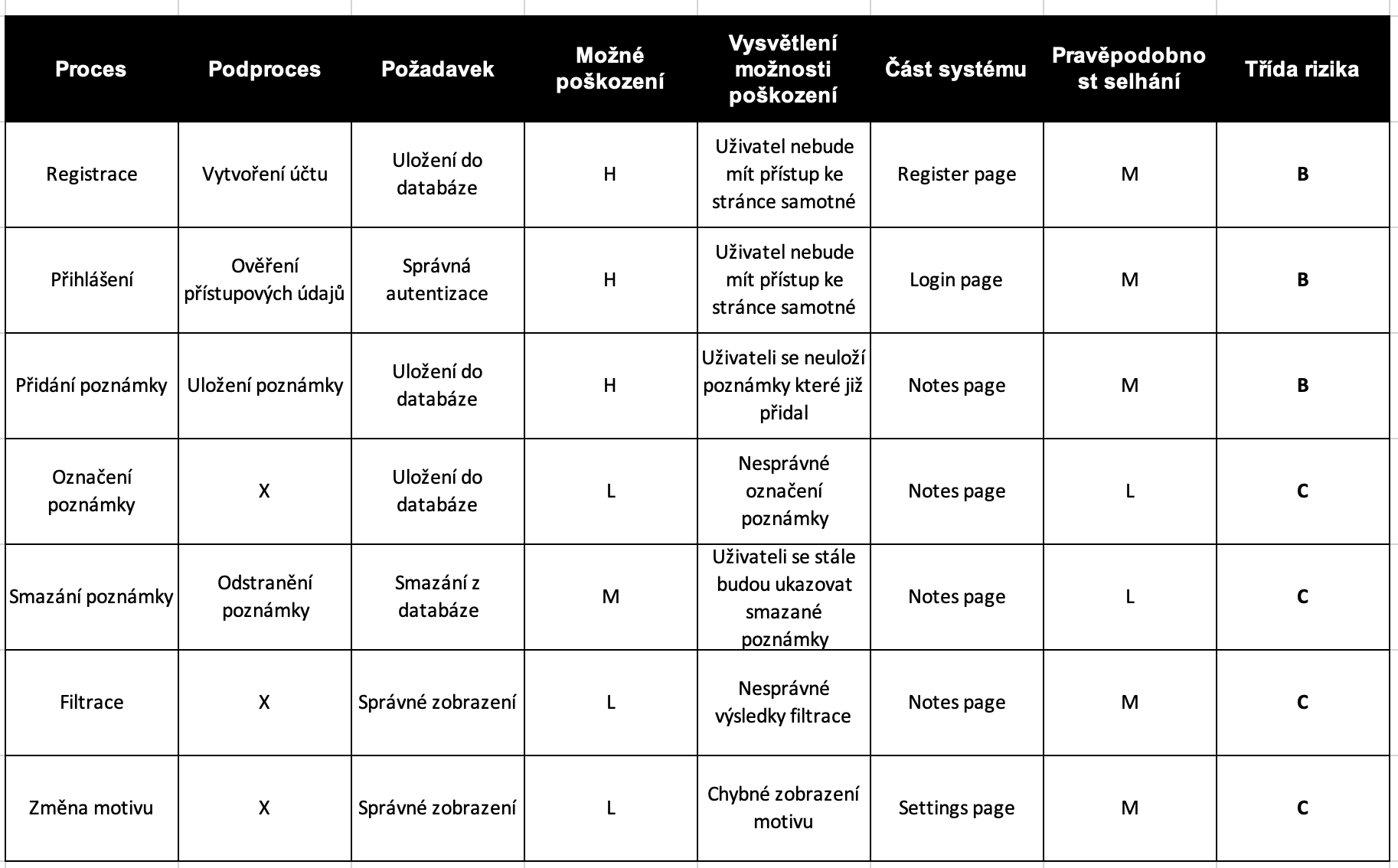
## Přehlednost částí aplikace:

Stránky které budeme testovat jsou:

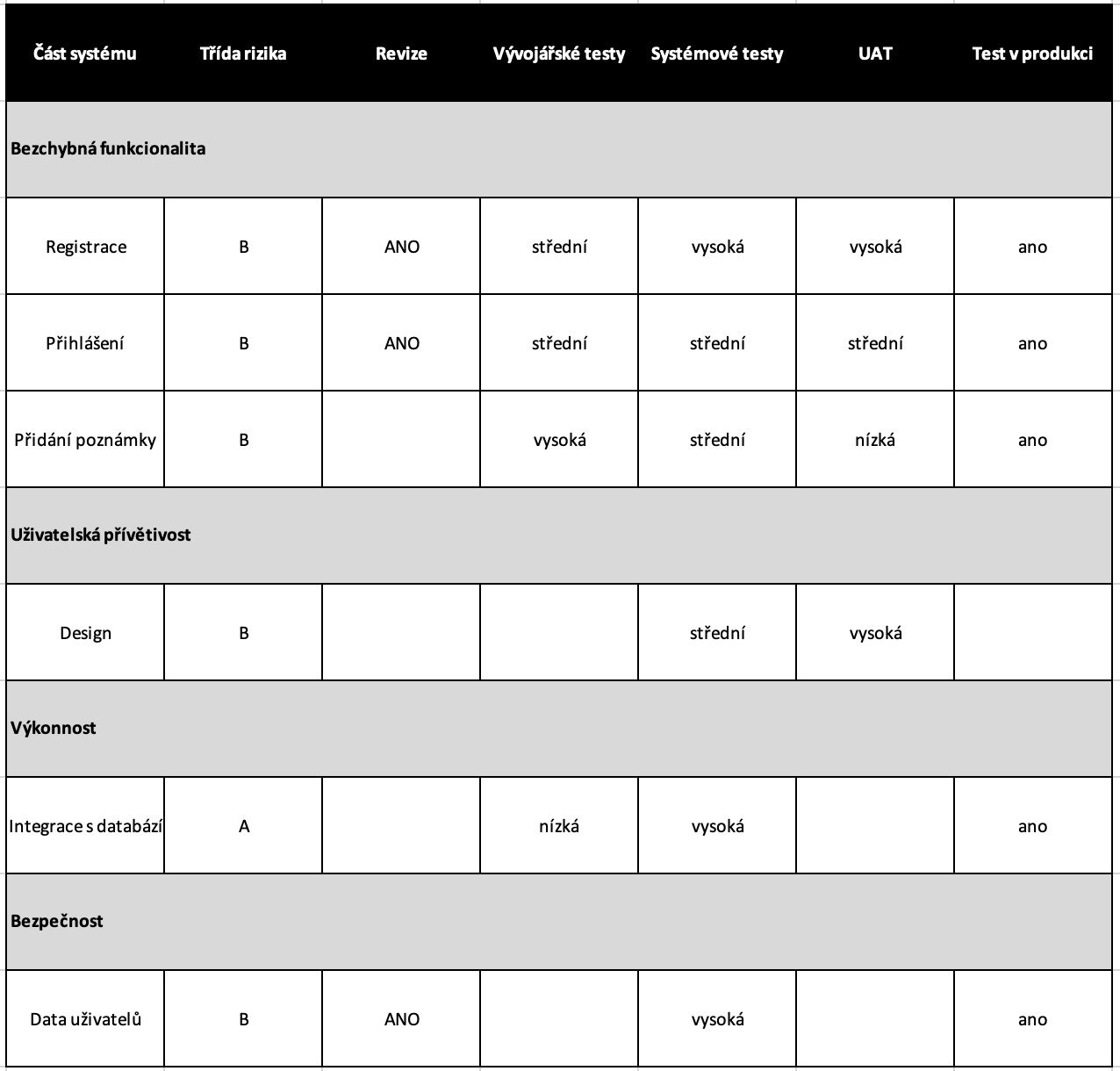
* Login page - test validních a nevalidních vstupů
* Register page - test validních a nevalidních vstupů
* Notes page - přidání a mazání poznámek, filtrace, stránkování
* Settings page - změna vzhledu stránky

## 

## Prioritizace částí aplikace:



## Test levels:



## Testy vstupů: Třídy ekvivalence

Registrační formulář

1. Uživatelské jméno
   * Nevalidní třída ekvivalence z technického pohledu
     + Uživatelské jméno kratší než 5 znaků
     + Uživatelské jméno obsahující speciální znaky
     + Uživatelské jméno obsahující mezery
     + Prázdné pole
   * Nevalidní třída ekvivalence z business pohledu
     + Uživatelské jméno už bylo použito
   * Validní třídy ekvivalence
     + Uživatelské jméno delší než 5 znaků z velkých nebo malých písmen
     + Uživatelské jméno delší než 5 znaků z číslic
     + Uživatelské jméno delší než 5 znaků z číslic a písmen
2. Email
   * Nevalidní třída ekvivalence z technického pohledu
     + Email nedodržuje platný formát
     + Prázdné pole
   * Nevalidní třída ekvivalence z business pohledu
     + Email už byl použit
   * Validní třídy ekvivalence
     + Validní email
3. Heslo
   * Nevalidní třída ekvivalence z technického pohledu
     + Heslo je kratší než 4 znaky
     + Prázdné pole
   * Nevalidní třída ekvivalence z business pohledu
     + Heslo nemá alespoň jedno číslo a jedno velké a malé písmeno
   * Validní třídy ekvivalence
     + Validní heslo
4. Potvrzení hesla
   * Nevalidní třída ekvivalence z technického pohledu
     + Potvrzení hesla se neshoduje s heslem
     + Prázdné pole
   * Validní třída ekvivalence
     + Potvrzení hesla je shodné s heslem

## Formulář na nové poznámky

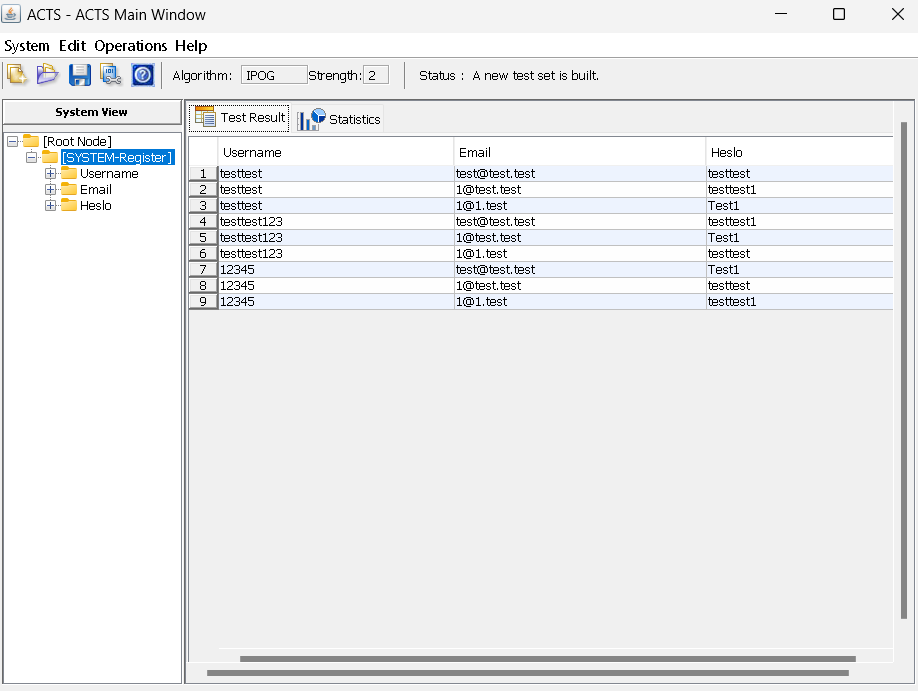
1. Jméno poznámky
   * Nevalidní třída ekvivalence z technického pohledu
     + Prázdné pole
   * Validní třída ekvivalence
     + Text z písmen, čísel a speciálních znaků
2. Datum
   * Nevalidní třída ekvivalence z technického pohledu
     + Prázdné pole
   * Nevalidní třída ekvivalence z business pohledu
     + Datum v minulosti
     + Neplatné datum
   * Validní třída ekvivalence
     + Dnešní datum
     + Datum v budoucnosti
3. Důležitost
   * Validní třída ekvivalence
     + Zaškrtnuté pole
     + Nezaškrtnuté pole
4. Popis poznámky
   * Nevalidní třída ekvivalence z technického pohledu
     + Prázdné pole
   * Validní třída ekvivalence
     + Text z písmen, čísel a speciálních znaků

## 

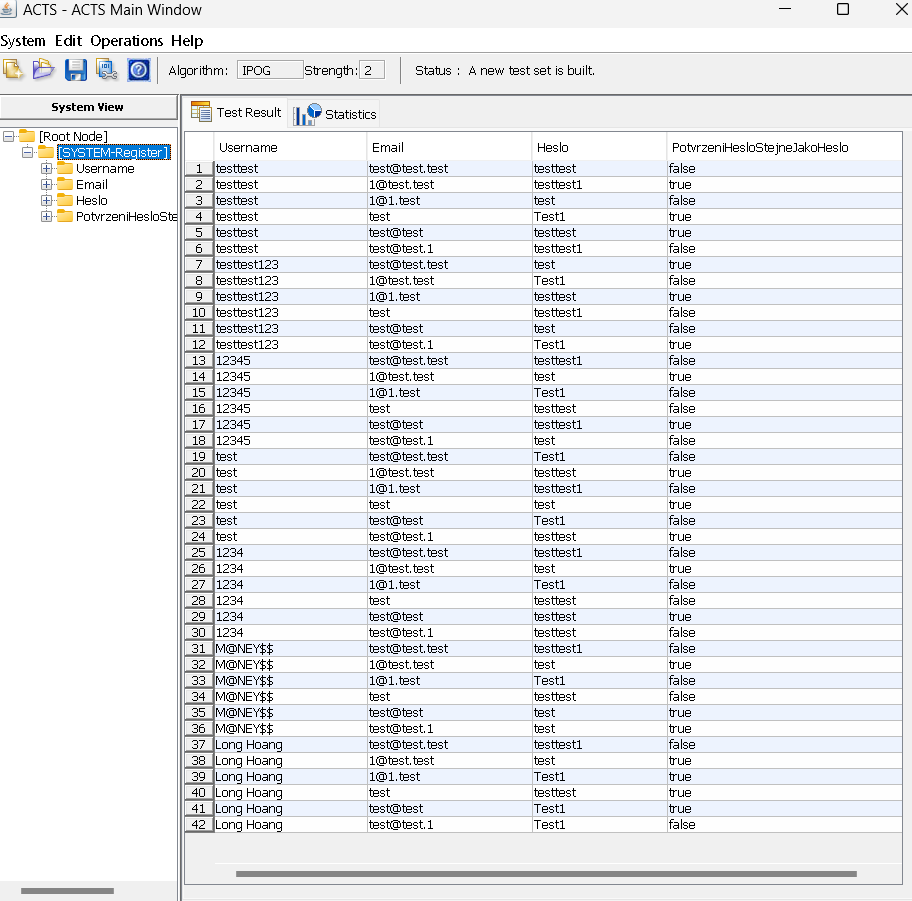
## Kombinace testovacích dat:

Registrační formulář

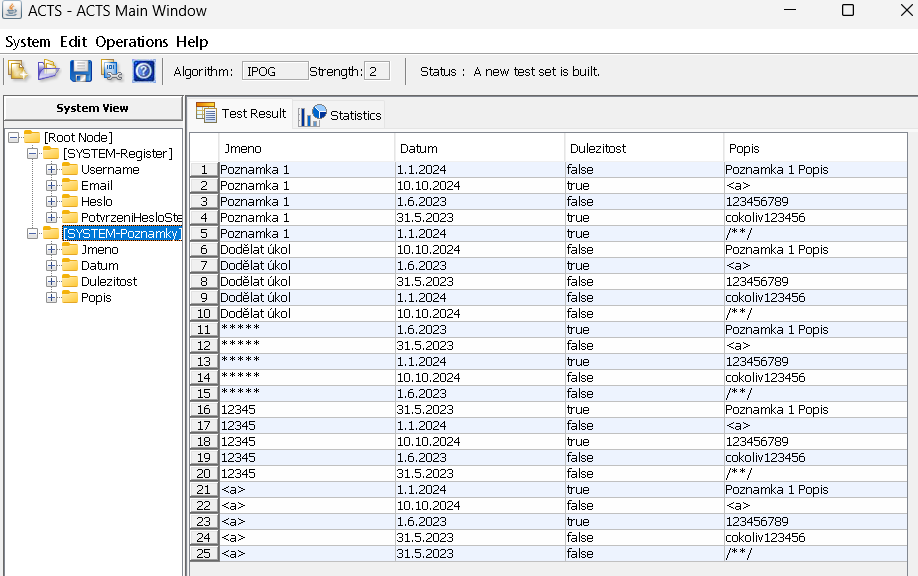
Kombinace dat s pouze validními vstupy



Kombinace dat s validními i nevalidními vstupy



## Formulář na nové poznámky

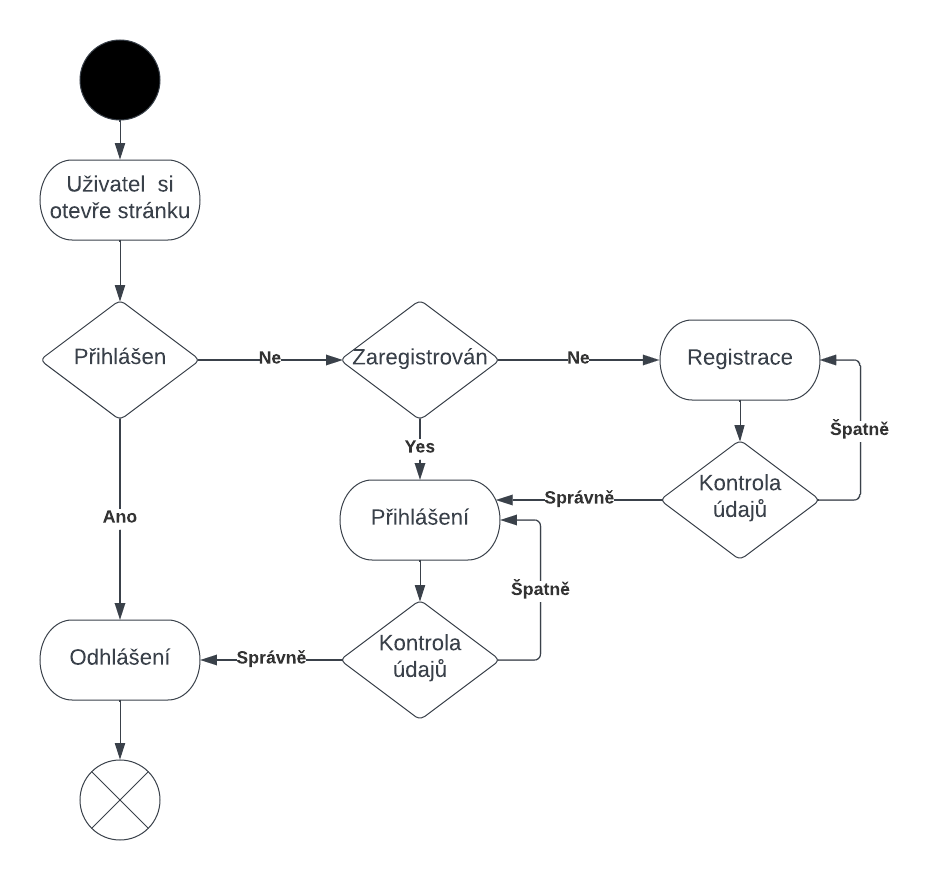


## 

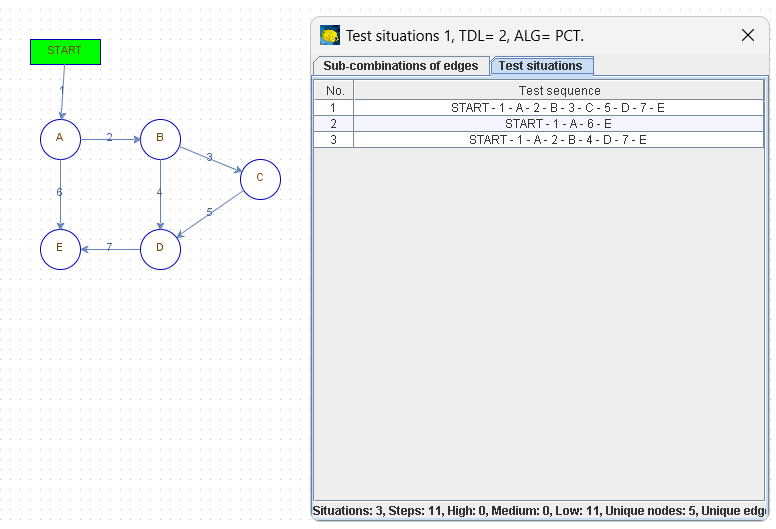
## Testy průchodů

Proces přihlášení, registrace a odhlášení:

Diagram

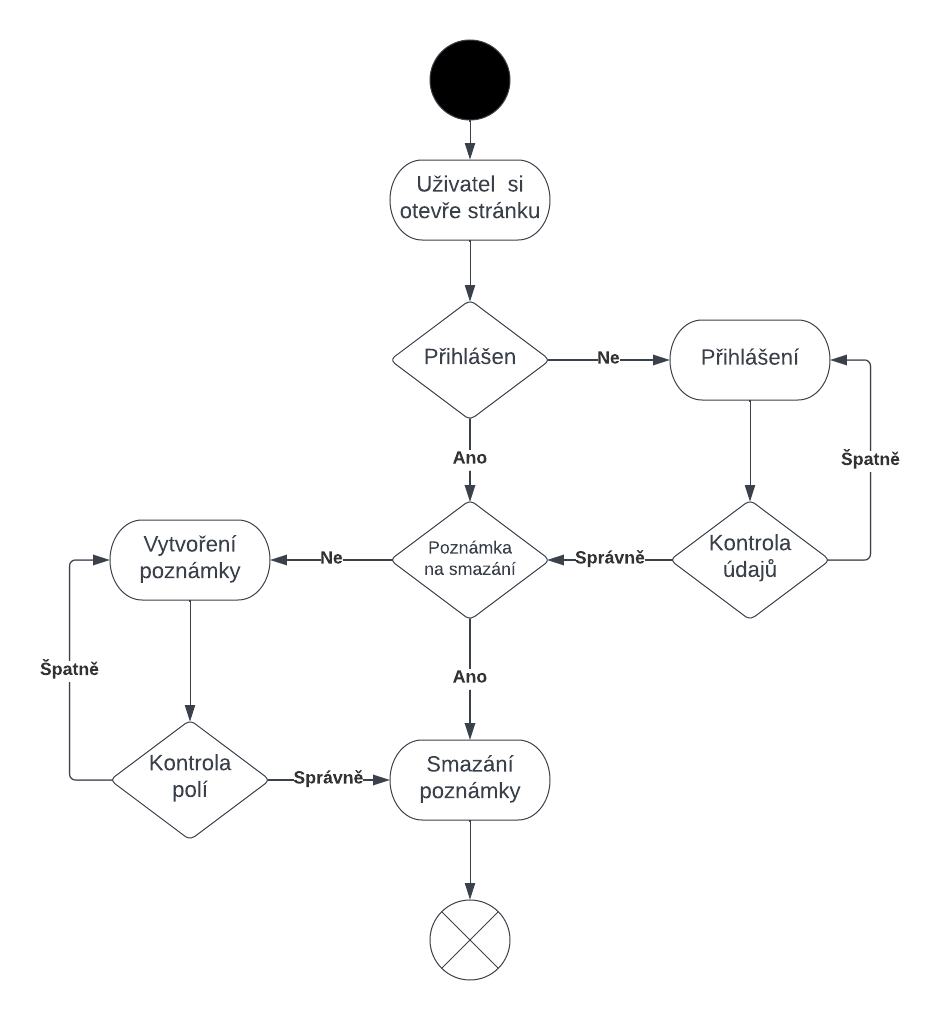


Procesní testy s TDL 2

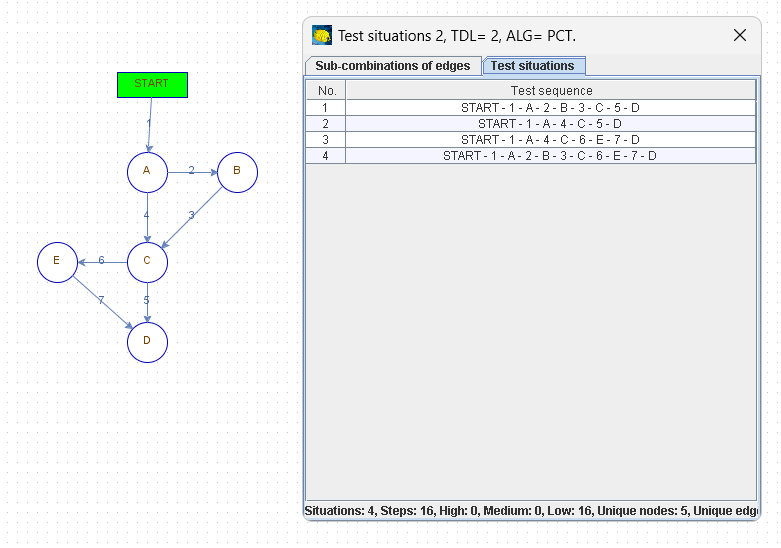


Proces mazání poznámky

Diagram



Procesní testy s TDL 2



## Testovací scénáře

### Test helloParametrizedTest:

* ID: PARAMTEST2
* Název testu: helloParametrizedTest
* Popis testu: Tento parametrizovaný test testuje zda-li se na úvodní stránce v navigačním panelu objeví správný text(“Vítej [jméno uživatele]”)
* Vstupní podmínky: Uživatel je zaregistrován, uživatel je schopen se přihlásit a mít přístup k úvodní stránce
* Testovací data:
  1. Uživatelské jméno:test1; Heslo:test1password
  2. Uživatelské jméno:test2; Heslo:test2password
  3. Uživatelské jméno:test3; Heslo:test3password
  4. Uživatelské jméno:test4; Heslo:test4password
* Očekávaný výsledek: Pro všechny testovací data se objeví správná hláška ve tvaru: “Vítej [Uživatelské jméno]”
* Stav přípravy:finální
* Kroky testovacího scénáře pro jedny testovací data:
  1. LoginPage
     + Popis kroku testu: Načtení stránky na přihlášení
     + Očekávaný výsledek kroku testu: Načtená stránka na přihlášení
  2. LoginInfo
     + Popis kroku testu: Zadání uživatelského jména a hesla do formuláře na přihlášení
     + Očekávaný výsledek kroku testu: Vyplněná políčka přihlašovacího formuláře
  3. Login
     + Popis kroku testu: Zmáčknutí tlačítka “Přihlásit se”
     + Očekávaný výsledek kroku testu: Zobrazí se domovská stránka
  4. CheckHello
     + Popis kroku testu: Zkontrolování hlášky v navigačním panelu
     + Očekávaný výsledek kroku testu: Uživatelské jméno v navigačním panelu se bude shodovat s přihlašovacím jménem

### Test countAllTasksTest:

* ID: COUNTTEST1
* Název testu: countAllTasks
* Popis testu: Tento test spočítá kolik má uživatel poznámek
* Vstupní podmínky: Uživatel je zaregistrován, uživatel je schopen se přihlásit, uživatel má přístup k úvodní stránce a stránce se všemi poznámkami, uživatel bude mít vytvořeno 20 poznámek
* Testovací data:
  1. Uživatelské jméno:test1; Heslo:test1password
* Očekávaný výsledek: Test spočítá poznámky a bude jich 20.
* Stav přípravy:finální
* Kroky testovacího scénáře:
  1. Login
     + Popis kroku testu: Přihlášení
     + Očekávaný výsledek kroku testu: Zobrazí se domovská stránka
  2. NotesPage
     + Popis kroku testu: Přejde na stránku s poznámkami
     + Očekávaný výsledek kroku testu: Zobrazí se stránka s poznámkami
  3. CountTasks
     + Popis kroku testu: Spočítá počet poznámek na první straně, pokud jich je 0, vrátí se počet poznámek. Pokud je poznámek více než 0, vybere poslední stránku v poznámkách na stránkování a spočítá počet poznámek na této stránce. Počet poznámek dopočítá přes počet stránek plných poznámek + počet poznámek na poslední straně
     + Očekávaný výsledek kroku testu: Počet poznámek by měl být roven 20