

Příručka administrátora Framework NET Genium

Obsah

1	Základní informace	5
1.1	Popis frameworku	5
1.2	Technické parametry frameworku	5
1.3	Technické parametry vývoje pomocí frameworku NET Genium	6
2	Přihlášení do NET Genia	7
3	Ovládací plocha	8
4	Aplikace	9
5	Aplikační skupiny	10
6	Editační formuláře	11
6.1	Ovládací prvky editačního formuláře	11
6.2	Oprávnění ke vstupu do editačního formuláře	14
6.3	Oprávnění k zobrazení obsahu editačního formuláře	15
6.4	Vlastnictví záznamů	15
6.5	Spoluvlastnictví záznamů	15
6.6	Oprávnění k zobrazení hodnoty ovládacího prvku v editačním formuláři	15
6.7	Oprávnění k uložení záznamu v editačním formuláři	16
6.8	Oprávnění ke smazání záznamu v editačním formuláři	16
6.9	Identifikátory ovládacích prvků editačního formuláře	16
6.10	Zamykání záznamů	16
6.11	Historie a sledování změn	17
6.12	Synchronizace databázových tabulek	17
7	Nahlížečské stránky	19
7.1	Ovládací prvky nahlížečské stránky	19
7.2	Oprávnění ke vstupu na nahlížečskou stránku	20
7.3	Oprávnění k zobrazení obsahu nahlížečské stránky	21
8	Kategorie	22
9	Portlety	23
10	Oblíbené položky	24
11	Picker	25

12 Databázový dotaz	26
12.1 Běžný dotaz	26
12.2 Agregáčn� dotaz	27
12.3 Vnořen� záznamy	28
12.4 Omezuj�c� podmínky dotazu	29
12.5 Připojení (joiny)	30
13 Proměnn� a identifik�tory	31
14 Serverov� funkce	32
15 Extern� funkce	33
16 Skript	34
16.1 Z�kladn� informace	34
16.2 Nov� z�znam vs existuj�c� z�znam v datab�zi	36
16.3 Skriptov� proměnn�	36
16.4 Uložen� hodnoty proměnn� do datab�zov�ho ovl�dac�ho prvku	37
16.5 Porovn�v�n� hodnot proměnn�ch	37
16.6 Konverze hodnot na jin� datov� typ	38
17 Javascript	39
18 Tisk	41
19 Aplikace „Nastaven�“	42
20 Nastaven� NET Genia	44
20.1 Nastaven� NET Genia konfigurovan� ve webov� aplikaci	44
20.2 Konfigura�n� soubory	44
20.3 Souborov� př�lohy	47
21 Reporty	48
22 Logov�n�	49
22.1 Logov�n� incidentů	49
22.1.1 Logov� soubory	49
22.1.2 SysLog	49
22.2 Logov�n� rychlosti webov� aplikace	50
23 Instalace	51
24 Utility	52

25 Zálohování	53
---------------------	----

1 Základní informace

1.1 Popis frameworku

NET Genium je framework určený pro Xtreme Rapid Development, se zaměřením na webové aplikace, které stojí na třech základních pilířích: na organizaci, vizualizaci a sdílení dat.

Organizace

Organizace znamená pořizování, modifikaci nebo mazání dat, dále prohledávání, filtrování, export nebo import dat, a to buď online ve webovém prohlížeči nebo offline pomocí nativní aplikace.

Vizualizace

Vizualizace znamená zobrazování dat v jedné z nejčastěji používaných forem, tj. v tabulce, nebo v kalendáři ala Outlook, v mapě, v ganttově diagramu, ve vícezdrojovém kalendáři nebo ve stromové struktuře, a dále znamená agregaci a následné zobrazování dat buď v běžné tabulce, kontingenční tabulce, nebo v grafu.

Sdílení

Sdílení znamená, že ke všem informacím mají uživatelé přístup online. Každý uživatel pracuje pod svým vlastním uživatelským účtem, který definuje jeho role či oprávnění, a zanechává za sebou digitální stopu, protože veškeré změny v datech, které provede, se zaznamenávají do historie.

Procesy, pracovní postupy a algoritmy

Komfort v práci s daty je standardem všech webových aplikací založených na frameworku NET Genium. Pro každého správce aplikace je ale nejzajímavější, že dokáže do své aplikace snadno implementovat specifické procesy, pracovní postupy nebo algoritmy, které jsou unikátní pro každého koncového zákazníka. Funkcionalita třech základních pilířů je zajištěna zcela automaticky, proto se může správce aplikace plně soustředit pouze na činnosti, které zajišťují přidanou hodnotu celé aplikace.

Informační systémy na míru a funkční prototypy

NET Genium se skvěle hodí na tvorbu jednoduchých i složitých informačních systémů na míru, nebo na přípravu funkčních prototypů aplikací, které se podle našich zkušeností výrazně neodlišují od finálního zadání. Díky NET Geniu vznikají robustní webové aplikace, které jsou stabilní, bezpečné a škálovatelné. Tyto aplikace mohou být provozovány v cloudu nebo na privátních či kancelářských serverech. Ke své činnosti vyžadují pouze webový prohlížeč s podporou javascriptu.

1.2 Technické parametry frameworku

- Webová aplikace napsaná v jazyce C# na platformě .NET
- Podpora databází MS SQL Server nebo Firebird
- Víceuživatelská aplikace s vlastním systémem uživatelských oprávnění nezávisle na IIS nebo Active Directory
- Integrovaný grafický návrhář určený k vývoji aplikací pro uživatele s administrátorským oprávněním
- Důsledná podpora většiny webových prohlížečů a responzivní design
- Bezpečnost v souladu s metodikou OWASP garantovaná několika nezávislými penetračními testy
- Optimalizovaný výkon pro méně výkonné nebo kancelářské servery

- Optimalizovaný výkon pro práci s velkým množstvím dat a fulltextové vyhledávání v databázových záznamech i souborových přílohách
- Podpora In-Memory tabulek
- Podpora single sign-on ve spojení s Active Directory
- Podpora běhu aplikace ve více instancích s rozdílným nastavením úrovně zabezpečení IIS
- Podpora běhu aplikace ve více kloncích – provozní, vývojové nebo archivní prostředí
- Nástroje pro vydávání nových release z vývojového do provozního nebo archivního prostředí
- Nástroje pro přesun časově omezených a zároveň konzistentních dat do archivního prostředí
- Nástroje pro exporty, importy nebo konverze velkého množství dat

1.3 Technické parametry vývoje pomocí frameworku NET Genium

- Skládání a propojování dvou základních stavebních jednotek aplikací – editačních formulářů a nahlížecích stránek
- Knihovna ovládacích prvků pro chytré tabulky, klasické kalendáře nebo plánovací vícezdrojové kalendáře, rozvrhy hodin, ganttovy diagramy, grafy, mapy nebo stromové struktury
- Podpora vícejazyčnosti
- Grafický návrhář databázových dotazů
- Grafický návrhář skriptů spouštěných na straně serveru
- Volání nejčastěji používaných funkcí pro práci s daty z knihovny funkcí
- Volání vlastních funkcí napsaných v jazyce C#
- Nástroje pro spouštění skriptů z naplánovaných úloh
- CodeMirror pro psaní JavaScriptu
- Podpora tisku do tiskových šablon XLS, DOC, TXT nebo HTML
- Automatická konverze tiskových sestav do PDF
- Synchronizace dat se zařízeními Android nebo iOS
- Symetrická synchronizace dat (multiple master) s libovolnou aplikací založenou na platformě NET Genium
- API pro komunikaci s libovolnou aplikací založenou na platformě .NET

2 Přihlášení do NET Genia

- NET Genium podporuje tři základní typy autorizace uživatelů:
 - **Běžný uživatel** – pracuje s daty, a má k dispozici pouze obsah a funkce na základě svých oprávnění
 - **Administrátor** – má přístup ke všem datům, a vytváří nebo modifikuje aplikace pomocí integrovaného návrháře
 - **Anonymní uživatel** – vyplňuje údaje do webových formulářů, které se díky iframům stávají součástí webových prezentací
- Licenční model je založený na maximálním počtu současně přihlášených uživatelů do aplikace. Celkový počet licencí může být kdykoliv rozšířen.
- Každé přihlášení vyžaduje jednu uživatelskou licenci. Z jedné IP adresy se může uživatel přihlásit do aplikace vícekrát, přesto bude čerpána pouze jedna licence. Anonymní uživatelé licenci nepotřebují.
- Výchozí instalace NET Genia obsahuje dvě základní uživatelské skupiny:
 - **Users** – běžní uživatelé
 - **Administrators** – administrátoři

3 Ovládací plocha

- Ovládací plocha se skládá ze tří základních částí:
 - **Záhlaví** – záhlaví tvoří tematicky laděný barevný pruh s volitelným logem, jméno aktuálně přihlášeného uživatele, počet celkově přihlášených uživatelů, aktuální datum, odkaz na odhlášení uživatele, a dále seznam dostupných aplikačních skupin
 - **Navigační oblast** – navigační oblast tvoří seznam dostupných aplikací a seznam oblíbených položek
 - **Hlavní okno** – hlavní okno zobrazuje tři druhy stránek:
 - Portlety neboli hlavní stránku či dashboard
 - Nahlížecké stránky
 - Editační formuláře
- Detailní popis ovládací plochy je uveden v samostatné příručce „Ovládací plocha“.

4 Aplikace

- NET Genium se skládá ze sady navzájem propojených aplikací.
- Každou aplikaci lze vytvořit ze dvou základních stavebních jednotek aplikací – editačních formulářů a nahlížecích stránek.
 - Editační formulář slouží k editaci jednoho databázového záznamu.
 - Nahlížecí stránka zobrazuje více záznamů najednou, a slouží k vizualizaci, prohlížení, filtrování a prohledávání databázových záznamů.
 - Nahlížecí stránka je zároveň prostředníkem pro vstup do editačního formuláře – buď za účelem vytvoření nového záznamu v databázi, nebo za účelem editace konkrétního již existujícího záznamu.
 - Editační formuláře a nahlížecí stránky mohou obsahovat tlačítka, která slouží k vyvolávání událostí. Takovou událostí může být právě i otevření dalšího editačního formuláře či nahlížecí stránky. Nejčastější způsob, jak z nahlížecí stránky otevřít editační formulář za účelem založení nového databázového záznamu, je umístění tlačítka do nahlížecí stránky, které vyvolá otevření editačního formuláře.
 - Aplikace, která obsahuje pouze editační formuláře, postrádá rozhraní, pomocí kterého lze do editačních formulářů vstoupit. Proto musí nejmenší aplikace obsahovat alespoň jednu nahlížecí stránku.
- Aplikace mohou vytvářet nebo modifikovat pouze administrátoři, tedy uživatelé patřící do skupiny administrátorů. Export a import aplikací může provádět pouze jeden uživatel s nejvyšším oprávněním – uživatel „Administrator“ s databázovým ID 1.
- Úpravy aplikací je možné provádět pouze v administrátorském režimu. Administrátorský režim je dostupný pouze administrátorům, kteří se mohou libovolně přepínat mezi uživatelským a administrátorským režimem.
- V administrátorském režimu se webová aplikace zobrazuje takřka stejně jako v uživatelském režimu, jen jsou navíc k dispozici navigační prvky a ovládací nástroje určené k úpravě aplikací či jejich částí.
- Úpravy aplikací se skládají z následujících činností:
 - vytváření nebo modifikace editačních formulářů,
 - vytváření nebo modifikace nahlížecích stránek,
 - vytváření nebo modifikace ovládacích prvků, ze kterých se skládají editační formuláře a nahlížecí stránky,
 - seskupování nahlížecích stránek do kategorií.
- Detailní popis nastavení aplikací je uveden v samostatné příručce „Aplikace“.

5 Aplikační skupiny

- Aplikace jsou pro lepší přehlednost seskupovány do aplikačních skupin. Aplikační skupiny obvykle obsahují aplikace se stejným či podobným zaměřením.
- Na úrovni aplikačních skupin je možné definovat konkrétní administrátory, kteří mohou aplikace uvnitř aplikační skupiny modifikovat. Ostatní administrátoři pak v těchto aplikacích nemohou provádět žádné změny, přestože jsou v administrátorském režimu.
- NET Genium automaticky slučuje aplikační skupiny s aplikacemi, pokud aplikační skupina obsahuje pouze jednu aplikaci, která má stejný název jako aplikační skupina.
- Detailní popis nastavení aplikačních skupin je uveden v samostatné příručce „Aplikační skupina“.

6 Editační formuláře

- Editační formulář je grafické rozhraní, které slouží k editaci jednoho databázového záznamu – buď za účelem vytvoření nového záznamu v databázi, nebo za účelem editace konkrétního již existujícího záznamu.
- Každý editační formulář má k sobě přidruženou jednu databázovou tabulku uvnitř v databázi, do které se ukládají samotná uživatelská data vyplňovaná uživatelem ve formuláři. Název databázové tabulky je odvozen z názvu editačního formuláře, a začíná prefixem „ng_“. Proces zakládání či modifikace přidružené databázové tabulky je zcela automatický, administrátoři tedy nikdy nemění samotnou databázovou tabulku uvnitř v databázi, ale pouze mění vlastnosti editačního formuláře nebo jeho ovládacích prvků.
- Každá přidružená databázová tabulka obsahuje následující databázové sloupce:
 - „id“ – primární klíč neboli jednoznačný identifikátor každého záznamu v databázové tabulce, který má buď databázový typ „int NOT NULL“ nebo „long NOT NULL“, v závislosti na nastavení editačního formuláře.
 - „system“ – pomocí hodnoty „True“ určuje systémový databázový záznam, který není možné smazat. Příkladem je záznam s ID 1 (Administrator) a 2 (Anonymous) v databázové tabulce uživatelů NET Genia, které mají zásadní charakter pro funkčnost celé aplikace.
 - „pid“ – ID rodičovského záznamu
 - „pform“ – ID editačního formuláře, ve kterém se nachází rodičovský záznam
 - „userid“ – ID uživatele, který je vlastníkem databázového záznamu, resp. záznam v databázi vytvořil
- Detailní popis nastavení editačních formulářů je uveden v samostatné příručce „Editační formulář“.

6.1 Ovládací prvky editačního formuláře

- Editační formulář se skládá z databázových a vizuálních ovládacích prvků.
 - Každý databázový ovládací prvek má k sobě přidružený jeden sloupec v přidružené databázové tabulce, do kterého se ukládají hodnoty vyplněné v editačním formuláři. Aby měl editační formulář nějaký smysl, musí obsahovat alespoň jeden databázový ovládací prvek.
 - Vizuální ovládací prvky mají ve formuláři pouze informativní charakter – jsou jimi například nadpis, nahlížecí tabulka, tlačítko, grafický oddělovač s horizontální linií nebo bez apod.
 - Každý editační formulář automaticky obsahuje ovládací prvek „ActionButtons“, který se skládá ze třech základních tlačítek potřebných pro práci s formulářem – „Uložit“, „Smazat“ a „Zpět“. Detailní popis ovládacího prvku „ActionButtons“ je uveden v samostatné příručce „ActionButtons – Ovládací prvek v editačním formuláři“.
- Ovládací prvky jsou v editačním formuláři řazeny pod sebou, případně mohou opticky tvořit více sloupců pomocí přichytávání ovládacích prvků k předchozím. Mezi databázovými ovládacími prvky je možné přeskakovat dopředu (dolů) pomocí klávesy „TAB“ a zpět (nahoru) pomocí kombinace kláves „SHIFT+TAB“.
- Mezi databázové ovládací prvky patří:
 - **TextBox** – textové pole libovolné šířky a výšky určené k zadávání prostého textu s maximální délkou 250 znaků u databáze Firebird a s neomezenou délkou u databáze MSSQL, bez možnosti vložení vynuceného přechodu na novou řádku – enteru. Při nastavení šířky na 0 pixelů se u textového pole nezobrazuje typické orámování, a také chybí možnost editace textu – hodnota textového pole se zobrazuje jako běžný odstavec s prostým textem. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „TextBox – Ovládací prvek v editačním formuláři“.

- **TextArea** – textové pole libovolné šířky a výšky určené k zadávání prostého textu s neomezenou délkou s možností vložení vynuceného přechodu na novou řádku – enteru. Při nastavení šířky na 0 pixelů se u textového pole nezobrazuje typické orámování, a také chybí možnost editace textu – hodnota textového pole se zobrazuje jako běžný odstavec s prostým textem. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „TextArea – Ovládací prvek v editačním formuláři“.
- **RichTextBox** – textové pole libovolné šířky a výšky určené k zadávání formátovaného textu s neomezenou délkou. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „RichTextBox – Ovládací prvek v editačním formuláři“.
- **ComboBox** – jednořádkový rozbalovací seznam libovolné šířky s možností výběru právě jedné hodnoty ze seznamu dostupných hodnot, případně s možností zadání vlastní hodnoty podobně jako u textového pole. Vybraná hodnota může mít maximální délku 250 znaků u databáze Firebird a neomezenou délku u databáze MSSQL. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „ComboBox – Ovládací prvek v editačním formuláři“.
- **ListBox** – víceřádkový seznam libovolné šířky a výšky s možností výběru právě jedné hodnoty ze seznamu dostupných hodnot. Vybraná hodnota může mít maximální délku 250 znaků u databáze Firebird a neomezenou délku u databáze MSSQL. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „ListBox – Ovládací prvek v editačním formuláři“.
- **MultiListBox** – víceřádkový seznam libovolné šířky a výšky s možností výběru více hodnot ze seznamu dostupných hodnot. Vybrané hodnoty jsou v databázi uloženy oddělené tabulátorem ve formě textového řetězce neomezené délky. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „MultiListBox – Ovládací prvek v editačním formuláři“.
- **Radio** – přepínač s možností výběru právě jedné hodnoty ze seznamu dostupných hodnot. Vybraná hodnota může mít maximální délku 250 znaků u databáze Firebird a neomezenou délku u databáze MSSQL. Přepínač může zobrazovat dostupné hodnoty buď vertikálně pod sebou, nebo horizontálně vedle sebe. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Radio – Ovládací prvek v editačním formuláři“.
- **CheckBox** – zaškrťovací tlačítko, které při zaškrtnutí ukládá do databáze předem stanovenou textovou hodnotu s maximální délkou 250 znaků u databáze Firebird a s neomezenou délkou u databáze MSSQL. V případě nezaškrtnutí ukládá do databáze prázdný textový řetězec. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „CheckBox – Ovládací prvek v editačním formuláři“.
- **ForeignKey** – jednořádkový rozbalovací seznam libovolné šířky s možností výběru právě jedné hodnoty ze seznamu dostupných hodnot, případně textové pole zobrazující právě jednu hodnotu ze seznamu dostupných hodnot. ForeignKey používá databázový typ „int“, a do databáze ukládá primární klíč (ID) vybraného databázového záznamu. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „ForeignKey – Ovládací prvek v editačním formuláři“.
- **File** – ovládací prvek určený k vložení souborové přílohy pomocí tlačítka „Procházet...“. Maximální velikost vkládané souborové přílohy může být omezena na předem stanovenou velikost. File používá databázový typ „int“, a do databáze ukládá ID souborové přílohy, která je fyzicky uložena na disku v adresáři „Files“, a popisné údaje s názvem souboru a jeho velikostí má uložené v samostatném záznamu v databázové tabulce „sfiles“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „File – Ovládací prvek v editačním formuláři“.
- **Image** – ovládací prvek určený k vložení obrázku buď pomocí tlačítka „Procházet...“, nebo pomocí klávesy „CTRL+V“ z clipboardu. Maximální velikost vkládaného obrázku může být omezena na předem stanovenou velikost. Image používá databázový typ „int“, a do databáze ukládá ID souborové přílohy, která je fyzicky uložena na disku v adresáři „Files“, a popisné údaje s názvem souboru a jeho velikostí má uložené v samostatném záznamu v databázové tabulce „sfiles“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Image – Ovládací prvek v editačním formuláři“.

- **DatePicker** – datumový picker určený k zadání data pomocí rozbalovacích seznamů se dnem, měsícem a rokem, případně i s textovým polem pro zadání času. DatePicker používá databázový typ „datetime“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „DatePicker – Ovládací prvek v editačním formuláři“.
- **Schedule** – ovládací prvek určený ke stanovení časového období od-do, který se skládá buď z textových polí „Od“ a „Do“ a zaškrtačovacího tlačítka „Od-Do“, nebo z datumových pickerů „Od“ a „Do“ a zaškrtačovacího tlačítka „Od-Do“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Schedule – Ovládací prvek v editačním formuláři“.
- **Watcher** – ovládací prvek určený ke sledování data vytvoření záznamu a data poslední změny v záznamu. Skládá se ze čtyř textových polí „Zadáno kdy“, „Zadáno kým“, „Změněno kdy“ a „Změněno kým“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Watcher – Ovládací prvek v editačním formuláři“.
- **AutoSum** – jednořádkové textové pole libovolné šířky, které slouží k automatickému vyhodnocení agregační funkce na předem definovaná data z databáze. Mezi tyto agregační funkce patří počet, suma, průměr, maximální a minimální hodnota. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „AutoSum – Ovládací prvek v editačním formuláři“.
- Mezi vizuální ovládací prvky patří:
 - **Title** – nadpis stylu „H1“ nebo „H2“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Title – Ovládací prvek v editačním formuláři“.
 - **Tab** – záložka. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Tab – Ovládací prvek v editačním formuláři“.
 - **RichText** – formátovaný text. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „RichText – Ovládací prvek v editačním formuláři“.
 - **DataGrid** – nahlížecí tabulka s možností vyhledávání a filtrování záznamů, a s možností exportů a importů dat. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „DataGrid – Ovládací prvek v editačním formuláři“.
 - **LiteDataGrid** – nahlížecí tabulka bez možnosti personalizace, vyhledávání a filtrování záznamů, a bez možnosti exportů a importů dat. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „LiteDataGrid – Ovládací prvek v editačním formuláři“.
 - **History** – historie změn. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „History – Ovládací prvek v editačním formuláři“.
 - **Calendar** – klasický kalendář ala Outlook. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Calendar – Ovládací prvek v editačním formuláři“.
 - **TimeTable** – rozvrh hodin/ganttův diagram. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „TimeTable – Ovládací prvek v editačním formuláři“.
 - **Planner** – vícezdrojový plánovací kalendář. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Planner – Ovládací prvek v editačním formuláři“.
 - **Chart** – graf. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Chart – Ovládací prvek v editačním formuláři“.
 - **Map** – mapa založená na mapových podkladech OpenStreetMap. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Map – Ovládací prvek v editačním formuláři“.
 - **Tree** – stromová struktura. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Tree – Ovládací prvek v editačním formuláři“.
 - **Button** – tlačítko určené k volání javascriptových funkcí ve webovém prohlížeči, skriptů na straně serveru, tisku databázových záznamů do tiskových šablon, přesměrování uživatele do konkrétního editačního formuláře nebo na

- konkrétní nahlížečící stránku. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Button – Ovládací prvek v editačním formuláři“.
- **HTML** – HTML kód. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „HTML – Ovládací prvek v editačním formuláři“.
 - **JavaScript** – javascriptový kód. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „JavaScript – Ovládací prvek v editačním formuláři“.
 - **Space** – grafický oddělovač s horizontální linií nebo bez. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Space – Ovládací prvek v editačním formuláři“.
- K ovládacím prvkům editačního formuláře je možné přistupovat pomocí javascriptu, a měnit tak jejich vlastnosti a chování. V případě databázových ovládacích prvků je možné měnit hodnoty textových polí, rozbalovacích seznamů atd. za předpokladu, že známe odkaz na ovládací prvek v objektovém modelu HTML dokumentu.

6.2 Oprávnění ke vstupu do editačního formuláře

- Do editačního formuláře lze vstoupit pouze pomocí navigačních prvků umístěných na nahlížečících stránkách nebo v editačních formulářích.
- Pro potřeby vytvoření nového záznamu v databázi může uživatel vstoupit do editačního formuláře následujícími způsoby:
 - Pomocí tlačítka umístěného na nahlížečící stránce nebo v editačním formuláři, které přesměruje uživatele do konkrétního editačního formuláře. Tento způsob se definuje přímo ve vlastnostech ovládacího prvku „Button“, nebo v jeho skriptu pomocí funkce „SCHEDULEFORM“.
 - Pomocí nahlížečící tabulky, která přesměruje uživatele do editačního formuláře, jehož data zobrazuje. Tento způsob se definuje přímo ve vlastnostech nahlížečící tabulky povolením zobrazení ikony „Nový záznam...“.
 - Pomocí ostatních vizuálních ovládacích prvků jako je klasický kalendář, plánovací vícezdrojový kalendář, rozvrh hodin, ganttův diagram nebo stromová struktura. Tento způsob se definuje přímo ve vlastnostech daného vizuálního ovládacího prvku povolením zadávání nových záznamů.
 - Pomocí nahlížečící stránky, která přesměruje uživatele do konkrétního editačního formuláře. Tento způsob se definuje přímo ve vlastnostech nahlížečící stránky.
- Pro potřeby editace konkrétního již existujícího záznamu v databázi může uživatel vstoupit do editačního formuláře pomocí následujících možností:
 - Pomocí nahlížečící tabulky, která zobrazuje na začátku každé své řádky ikonu tužky s funkcí otevření zobrazovaného databázového záznamu.
 - Pomocí ostatních vizuálních ovládacích prvků jako je klasický kalendář, plánovací vícezdrojový kalendář, rozvrh hodin, ganttův diagram, mapa nebo stromová struktura. Tyto ovládací prvky buď zobrazují ikonu tužky, nebo umožňují otevření databázového záznamu kliknutím na jeho zobrazované detaily.
 - Pomocí tlačítka umístěného na nahlížečící stránce nebo v editačním formuláři, které přesměruje uživatele na konkrétní databázový záznam editačního formuláře. Tento způsob se definuje ve skriptu ovládacího prvku „Button“ pomocí funkce „SCHEDULEFORM“.
 - Pomocí nahlížečící stránky, která přesměruje uživatele do prvního záznamu daného editačního formuláře. Tento způsob se definuje přímo ve vlastnostech nahlížečící stránky.

6.3 Oprávnění k zobrazení obsahu editačního formuláře

- Editační formulář jako jediný nemá možnost nastavení práv na zobrazení formuláře. Oprávnění k zobrazení obsahu editačního formuláře se řídí podle nastavení práv jeho vnitřních ovládacích prvků. Editační formulář se zobrazuje ve chvíli, kdy obsahuje alespoň jeden ovládací prvek, na který má aktuálně přihlášený uživatel právo. V opačném případě se v editačním formuláři zobrazuje pouze přihlašovací okno stejně jako na hlavní stránce.
- Každý ovládací prvek editačního formuláře má možnost nastavení uživatelských skupin nebo konkrétních uživatelů, kteří mají na ovládací prvek právo. Ostatní uživatelé ovládací prvek v editačním formuláři nevidí, a nemají možnost vyplnit jeho hodnotu, pokud se jedná o databázový ovládací prvek.
- Vytvoří-li uživatel prostřednictvím editačního formuláře v databázi nový záznam, ve všech databázových sloupcích, na které nemá právo, bude vyplněna databázová hodnota „null“.

6.4 Vlastnictví záznamů

- Každá databázová tabulka obsahuje sloupec „userid“ s ID uživatele, který je vlastníkem databázového záznamu, resp. záznam v databázi vytvořil.
- Databázové záznamy, které byly importovány nebo vytvořeny externí aplikací, nemusí mít hodnotu sloupce „userid“ vyplněnou. To znamená, že takové záznamy dosud vlastníka nemají. Vlastníkem se pak stane ten uživatel, který záznam otevře v editačním formuláři, a jako první uloží.

6.5 Spoluvlastnictví záznamů

- Databázové záznamy mohou mít kromě svého vlastníka také několik spoluvlastníků. Těmi mohou být jak jednotliví uživatelé, tak i uživatelské skupiny. Informace o spoluvlastníkovi není nastavena u každého databázového záznamu zvlášť stejně jako u vlastníka záznamu, ale nastavuje se pro příslušnou databázovou tabulku (editační formulář) jako celek.
- Vlastnictví nebo alespoň spoluvlastnictví záznamu je vyžadováno při editaci již existujícího záznamu v databázi a pokusu o jeho uložení v editačním formuláři. Není-li tedy uživatel vlastníkem nebo alespoň spoluvlastníkem existujícího záznamu, nebude mít možnost takový záznam v databázi změnit.
- Spoluvlastnictví záznamů nemá vliv na vytváření nových záznamů. Vlastníkem nově vytvořeného záznamu se stane vždy uživatel, který záznam v editačním formuláři uložil.

6.6 Oprávnění k zobrazení hodnoty ovládacího prvku v editačním formuláři

- Viditelnost hodnoty databázového ovládacího prvku může být řízena vlastnictvím či spoluvlastnictvím záznamu na úrovni jednotlivých databázových záznamů.
- Právo na zobrazení hodnoty databázového ovládacího prvku se nastavuje přímo ve vlastnostech ovládacího prvku.
- Ovládací prvek může být nastaven dvěma různými způsoby:
 - Všem uživatelům (výchozí nastavení)
 - Pouze vlastníkovi nebo spoluvlastníkovi záznamu. V takovém případě má oprávnění na zobrazení hodnoty pouze vlastník nebo spoluvlastník databázového záznamu. Ostatní uživatelé ovládací prvek v editačním formuláři nevidí, a nemají možnost zobrazit nebo vyplnit jeho hodnotu.

6.7 Oprávnění k uložení záznamu v editačním formuláři

- Oprávnění na uložení databázového záznamu je podmíněno oprávněním k editaci, resp. vlastnictvím nebo spoluvlastnictvím záznamu. Právo na editaci se nastavuje přímo ve vlastnostech editačního formuláře.
- Editací formulář může být nastaven dvěma různými způsoby:
 - Oprávnění k editaci má vlastník nebo spoluvlastník záznamu
 - Oprávnění k editaci má pouze spoluvlastník záznamu (výchozí nastavení)

6.8 Oprávnění ke smazání záznamu v editačním formuláři

- Oprávnění ke smazání databázového záznamu je podmíněno oprávněním k editaci, resp. vlastnictvím nebo spoluvlastnictvím záznamu, a současně oprávněním ke všem databázovým ovládacím prvkům editačního formuláře.
- Uživatel může smazat záznam v editačním formuláři,
 - má-li oprávnění k editaci databázového záznamu a zároveň
 - má právo na všechny databázové ovládací prvky editačního formuláře.
- Uživatel nemůže smazat záznam v editačním formuláři,
 - nemá-li oprávnění k editaci databázového záznamu nebo
 - vyskytuje-li se v editačním formuláři alespoň jeden databázový ovládací prvek, na který nemá právo.

6.9 Identifikátory ovládacích prvků editačního formuláře

- Identifikátor je odkaz na databázový ovládací prvek editačního formuláře uvnitř objektového modelu HTML dokumentu, a zároveň určuje název databázového sloupce, do kterého se v databázi ukládají hodnoty zadané prostřednictvím ovládacího prvku v editačním formuláři.
- Identifikátor je odvozen od názvu ovládacího prvku. Začíná prefixem „ng_“, a dále obsahuje alfanumerické znaky použité v názvu ovládacího prvku (systémové ovládací prvky prefix „ng_“ neobsahují).
- Identifikátor ovládacího prvku je k dispozici na záložce „Administrace“ přímo ve vlastnostech ovládacího prvku.
- Pomocí identifikátoru je možné přistupovat k databázovému ovládacímu prvků v javascriptu, a měnit vlastnosti, chování nebo hodnotu ovládacího prvku.
- Pomocí identifikátoru je také možné přistupovat k hodnotě uložené v databázovém ovládacím prvků
 - v javascriptu ve webovém prohlížeči,
 - ve skriptu na straně serveru,
 - v podmínkách nebo joinech databázových dotazů,
 - v ovládacím prvků „HTML“,
 - v tiskových šablonách.

6.10 Zamykání záznamů

- Databázové záznamy otevřené v editačním formuláři mohou být nastaveny tak, aby je v jednu chvíli mohl editovat pouze jeden uživatel.
- Zámky databázových záznamů se definují přímo v nastavení editačního formuláře zadáním časového intervalu v minutách, během kterého má být zámek aktivní. Po uplynutí tohoto intervalu se zámek automaticky uvolní.

- Uživatel, který databázový záznam otevře jako první, záznam automaticky zamyká. Během práce v editačním formuláři se zámek automaticky prodlužuje, dokud uživatel editační formulář neopustí. Při opuštění editačního formuláře se zámek automaticky uvolní.
- Zamčené záznamy mají v nahlížečské tabulce místo ikony lupy, která slouží k otevření databázového záznamu, ikonu zámku. Ikona zámku obsahuje tooltip s informací, kdo záznam uzamknul, a do kdy je zámek aktivní. I tak je však možné databázový záznam pomocí ikony zámku otevřít, ostatním uživatelům se editační formulář se zamčeným záznamem zobrazuje pouze ke čtení bez možnosti editace.

6.11 Historie a sledování změn

- Každý editační formulář má k sobě přidruženou ještě jednu databázovou tabulku uvnitř v databázi, do které se ukládají změny prováděné uživateli v jednotlivých databázových záznamech. Název této databázové tabulky je odvozen z názvu editačního formuláře, začíná prefixem „ng_“, a končí postfixem „_history“. Proces zakládání či modifikace této databázové tabulky je stejně jako u té hlavní zcela automatický.
- Databázová struktura tabulky s historií je stejná jako u hlavní tabulky. Navíc ale obsahuje následující sloupce:
 - „date“ – datum změny
 - „userid“ – přihlašovací jméno uživatele, který změnu provedl
 - „action“ – ID skriptu, pomocí kterého byla změna provedena
 - „changedid“ – ID záznamu, ke kterému se změna vztahuje
 - „created“ – hodnota „x“, pokud jde o vytvoření nového databázového záznamu
 - „deleted“ – hodnota „x“, pokud jde o smazání databázového záznamu
- Vytvoření nového záznamu v hlavní databázové tabulce zajistí vytvoření jednoho záznamu v tabulce s historií, kde jsou všechny hodnoty sloupců vyplněny stejně jako u hlavní tabulky. Navíc jsou vyplněny hodnoty sloupců „date“, „userid“, „action“, „changedid“ a „created“.
- Editace existujícího záznamu v hlavní databázové tabulce zajistí vytvoření jednoho záznamu v tabulce s historií, kde jsou vyplněny pouze hodnoty sloupců, u kterých dochází ke změně. Navíc jsou vyplněny hodnoty sloupců „date“, „userid“, „action“ a „changedid“. U ostatních sloupců bude v databázi vyplněna databázová hodnota „null“.
- Smazání existujícího záznamu v hlavní databázové tabulce zajistí vytvoření jednoho záznamu v tabulce s historií, kde jsou všechny hodnoty sloupců vyplněny stejně jako u hlavní tabulky těsně přes smazáním záznamu. Navíc jsou vyplněny hodnoty sloupců „date“, „userid“, „action“, „changedid“ a „deleted“.
- Ve výchozím nastavení je protokolování historie záznamů zapnuté, podle potřeby je však možné v nastavení editačního formuláře protokolování historie záznamů vypnout.

6.12 Synchronizace databázových tabulek

- Synchronizace slouží k udržování věrné kopie databázové tabulky v několika dalších databázích na vzdálených instalacích NET Genia najednou, např. na jiném serveru. Jde o symetrickou synchronizaci dat (multiple master), kdy změna v jednom exempláři databáze vyvolá požadavek na provedení stejné změny v dalších exemplářích. Požadavky jsou na serveru, který změnu inicioval, řazeny do fronty a na vzdálených serverech prováděny ve stejném pořadí.
- Synchronizace dat může probíhat jednosměrně nebo obousměrně. Jednosměrná synchronizace přenáší data ze zdrojové databáze do cílové pouze v jednom směru. Aby byla při jednosměrné synchronizaci zajištěna integrita dat, nesmí v cílové databázi docházet ke změnám záznamů obvykle vyvolaným uživateli NET Genia. Dochází-li ke změnám i v cílové databázi, musí být zvolena obousměrná synchronizace dat.
- Detailní popis nastavení synchronizace je uveden v příručce „Editační formulář“.

7 Nahlížečící stránky

- Nahlížečící stránka je grafické rozhraní, které slouží k vizualizaci, prohlížení, filtrování a prohledávání databázových záznamů.
- Detailní popis nastavení nahlížečících stránek je uveden v samostatné příručce „Nahlížečící stránka“.

7.1 Ovládací prvky nahlížečící stránky

- Nahlížečící stránka se skládá z vizuálních ovládacích prvků, které jsou řazeny pod sebou.
- Mezi vizuální ovládací prvky patří:
 - **TextBox** – jednořádkové textové pole libovolné šířky určené k zadávání prostého textu bez možnosti vložení vynuceného přechodu na novou řádku – enteru. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „TextBox – Ovládací prvek na nahlížečící stránce“.
 - **RichTextBox** – textové pole libovolné šířky a výšky určené k zadávání formátovaného textu. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „RichTextBox – Ovládací prvek na nahlížečící stránce“.
 - **Title** – nadpis stylu „H1“ nebo „H2“. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Title – Ovládací prvek na nahlížečící stránce“.
 - **Tab** – záložka. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Tab – Ovládací prvek na nahlížečící stránce“.
 - **RichText** – formátovaný text. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „RichText – Ovládací prvek na nahlížečící stránce“.
 - **DataGrid** – nahlížečící tabulka s možností vyhledávání a filtrování záznamů, a s možností exportů a importů dat. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „DataGrid – Ovládací prvek na nahlížečící stránce“.
 - **DataSet** – vícezdrojová nahlížečící tabulka s možností vyhledávání a filtrování záznamů. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „DataSet – Ovládací prvek na nahlížečící stránce“.
 - **History** – historie změn. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „History – Ovládací prvek na nahlížečící stránce“.
 - **Calendar** – klasický kalendář ala Outlook. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Calendar – Ovládací prvek na nahlížečící stránce“.
 - **TimeTable** – rozvrh hodin/ganttův diagram. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „TimeTable – Ovládací prvek na nahlížečící stránce“.
 - **Planner** – vícezdrojový plánovací kalendář. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Planner – Ovládací prvek na nahlížečící stránce“.
 - **Chart** – graf. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Chart – Ovládací prvek na nahlížečící stránce“.
 - **Map** – mapa založená na mapových podkladech OpenStreetMap. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Map – Ovládací prvek na nahlížečící stránce“.
 - **RDF/RSS News** – RSS feed. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „RDF/RSS News – Ovládací prvek na nahlížečící stránce“.
 - **Tree** – stromová struktura. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Tree – Ovládací prvek na nahlížečící stránce“.

- **E-Mail** – jednoduchý e-mailový klient pro doručenu poštu, který zobrazuje (avšak nestahuje ani neukládá do databáze) příchozí e-mailové zprávy z poštovního serveru POP3 nebo IMAP. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „E-Mail – Ovládací prvek na nahlížecí stránce“.
- **Button** – tlačítko určené k volání javascriptových funkcí ve webovém prohlížeči, skriptů na straně serveru, tisku databázových záznamů do tiskových šablon, přesměrování uživatele do konkrétního editačního formuláře nebo na konkrétní nahlížecí stránku. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Button – Ovládací prvek na nahlížecí stránce“.
- **HTML** – HTML kód. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „HTML – Ovládací prvek na nahlížecí stránce“.
- **JavaScript** – javascriptový kód. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „JavaScript – Ovládací prvek na nahlížecí stránce“.
- **Space** – grafický oddělovač s horizontální linií nebo bez. Detailní popis nastavení ovládacího prvku je uveden v samostatné příručce „Space – Ovládací prvek na nahlížecí stránce“.
- Nahlížecí stránka je zároveň prostředníkem pro vstup do editačního formuláře – buď za účelem založení nového záznamu v databázi, nebo za účelem editace konkrétního již existujícího záznamu.
 - Pro potřeby založení nového záznamu v databázi může uživatel vstoupit do editačního formuláře z nahlížecí stránky následujícími způsoby:
 - Pomocí tlačítka, které přesměruje uživatele do konkrétního editačního formuláře. Tento způsob se definuje přímo ve vlastnostech ovládacího prvku „Button“, nebo v jeho skriptu pomocí funkce „SCHEDULEFORM“.
 - Pomocí nahlížecí tabulky, která přesměruje uživatele do editačního formuláře, jehož data zobrazuje. Tento způsob se definuje přímo ve vlastnostech nahlížecí tabulky povolením zobrazení ikony „Nový záznam...“.
 - Pomocí ostatních vizuálních ovládacích prvků jako je klasický kalendář, plánovací vícezdrojový kalendář, rozvrh hodin, ganttův diagram nebo stromová struktura. Tento způsob se definuje přímo ve vlastnostech daného vizuálního ovládacího prvku povolením zadávání nových záznamů.
 - Pomocí nahlížecí stránky, která přesměruje uživatele do konkrétního editačního formuláře. Tento způsob se definuje přímo ve vlastnostech nahlížecí stránky.
 - Pro potřeby editace konkrétního již existujícího záznamu v databázi může uživatel vstoupit do editačního formuláře pomocí následujících možností:
 - Pomocí nahlížecí tabulky, která zobrazuje na začátku každé své řádky ikonu tužky s funkcí otevření zobrazovaného databázového záznamu.
 - Pomocí ostatních vizuálních ovládacích prvků jako je klasický kalendář, plánovací vícezdrojový kalendář, rozvrh hodin, ganttův diagram, mapa nebo stromová struktura. Tyto ovládací prvky buď zobrazují ikonu tužky, nebo umožňují otevření databázového záznamu kliknutím na jeho zobrazované detaily.
 - Pomocí tlačítka, které přesměruje uživatele na konkrétní databázový záznam editačního formuláře. Tento způsob se definuje ve skriptu ovládacího prvku „Button“ pomocí funkce „SCHEDULEFORM“.
 - Pomocí nahlížecí stránky, která přesměruje uživatele do prvního záznamu daného editačního formuláře. Tento způsob se definuje přímo ve vlastnostech nahlížecí stránky.

7.2 Oprávnění ke vstupu na nahlížecí stránku

- Na nahlížecí stránku lze vstoupit z navigační oblasti se seznamem dostupných aplikací nebo oblíbených položek, a dále pomocí navigačních prvků umístěných na hlavní stránce, na nahlížecích stránkách nebo v editačních formulářích.

- Každá nahlížečská stránka má možnost nastavení uživatelských skupin nebo konkrétních uživatelů, kteří mají na nahlížečskou stránku právo. Ostatní uživatelé nahlížečskou stránku v navigační oblasti nevidí, a nemají možnost do ní vstoupit.

7.3 Oprávnění k zobrazení obsahu nahlížečské stránky

- Nahlížečská stránka se zobrazuje ve chvíli, kdy obsahuje alespoň jeden ovládací prvek, na který má aktuálně přihlášený uživatel právo. V opačném případě se na nahlížečské stránce zobrazuje pouze přihlašovací okno stejně jako na hlavní stránce.
- Každý ovládací prvek nahlížečské stránky má možnost nastavení uživatelských skupin nebo konkrétních uživatelů, kteří mají na ovládací prvek právo. Ostatní uživatelé ovládací prvek na nahlížečské stránce nevidí.

8 Kategorie

- Nahlížecí stránky mohou být pro lepší přehlednost seskupovány do kategorií. Kategorie obvykle obsahují nahlížecí stránky se stejným či podobným zaměřením.
- Kategorie se definují uvnitř aplikací, a v navigační oblasti se zobrazují na stejné úrovni jako nahlížecí stránky umístěné přímo v aplikaci, tj. na stejné úrovni jako nahlížecí stránky, které nejsou kategorizované.
- Kategorie se zobrazují na prvním místě před nahlížecími stránkami, podobně jako je tomu v adresářové struktuře na disku u adresářů a souborů.
- Detailní popis nastavení kategorií je uveden v samostatné příručce „Kategorie“.

9 Portlety

- Portlety jsou zjednodušené vizuální ovládací prvky zobrazované na hlavní stránce nazývané též jako dashboard, a umožňují přehledné zobrazení rozdílných informací na jedné obrazovce.
- Portlety je možné zobrazovat v jednom, ve dvou nebo ve třech sloupcích – každý uživatel NET Genia má možnost si portlety nastavit podle svého uvážení.
- V portletech se zobrazují zjednodušené varianty následujících ovládacích prvků, umístěných na nahlížecích stránkách:
 - **RichText** – formátovaný text
 - **DataGrid** – nahlížecí tabulka
 - **Chart** – graf
 - **E-Mail** – jednoduchý e-mailový klient
 - **HTML** – HTML kód
- Nahlížecí tabulky nebo grafy zobrazované v portletech by měly respektovat omezenou šířku obrazovky vzhledem k možnosti uspořádat portlety do třech sloupců.
- Výchozí nastavení portletů definuje uživatel s nejvyšším oprávněním – uživatel „Administrator“ s databázovým ID 1. A to zvláště pro běžné uživatele, a zvláště pro anonymní uživatele.
- Běžný uživatel, který se přihlásí do NET Genia a nemá nastavené portlety, automaticky přebírá nastavení portletů od uživatele „Administrator“, resp. těch, na které má právo. Portlety si může dále přizpůsobit zcela samostatně.
- Anonymní uživatel a uživatelé ve skupině „External Users“ nemají právo na přizpůsobení svých portletů. Toto nastavení jim mohou měnit pouze uživatelé patřící do skupiny administrátorů.
- Detailní popis nastavení portletů je uveden v samostatné příručce „Portlety“.

10 Oblíbené položky

- Oblíbené položky jsou odkazy na často používané nahlížecké stránky, a zobrazují se v navigační oblasti, aby byly kdykoliv k dispozici.
- Výchozí nastavení oblíbených položek definuje uživatel s nejvyšším oprávněním – uživatel „Administrator“ s databázovým ID 1. A to zvláště pro běžné uživatele, a zvláště pro anonymní uživatele.
- Běžný uživatel, který se přihlásí do NET Genia a nemá nastavené oblíbené položky, automaticky přebírá nastavení oblíbených položek od uživatele „Administrator“, resp. těch, na které má právo. Oblíbené položky si může dále přizpůsobit zcela samostatně.
- Anonymní uživatel a uživatelé ve skupině „External Users“ nemají právo na přizpůsobení svých oblíbených položek. Toto nastavení jim mohou měnit pouze uživatelé patřící do skupiny administrátorů.
- Detailní popis nastavení oblíbených položek je uveden v samostatné příručce „Oblíbené položky“.

11 Picker

- Picker je dialogové okno, které zobrazuje nahlížecí tabulku s databázovými záznamy, jejichž hodnoty je možné převzít či zkopírovat do právě otevřeného editačního formuláře. Picker se typicky používá pro kopírování celých databázových záznamů nebo pro převzetí hodnoty cizího klíče (ForeignKey).
- Odkaz na dialogové okno pickeru s nahlížecí tabulkou a databázovými záznamy má podobu ikony lupy, a zobrazuje se v editačním formuláři napravo vedle ovládacího prvku, ke kterému je picker přidružen.
- Picker má i podobu ajaxového našeptávače, který začne nabízet databázové záznamy už při zadávání hodnoty do textového pole v editačním formuláři. Aby fungovalo ajaxové našeptávání, je nutné povolit vyhledávání hodnot zadaných v editačním formuláři.
- Nastavení pickeru se provádí v samostatném dialogu. Odkaz na tento dialog má podobu ikony lupy, a nachází se v levém dolním rohu dialogu nastavení ovládacího prvku, ke kterému je picker přidružen.
- Detailní popis nastavení pickeru je součástí příručky každého databázového ovládacího prvku.

12 Databázový dotaz

- Databázový dotaz definuje zdroj dat načítaných z databáze, která se následně používají nebo zobrazují v ovládacích prvcích editačních formulářů a nahlížecích stránek, nebo ve skriptech.
- Databázový dotaz určuje, jaké záznamy budou zobrazeny v nahlížecích tabulkách, portletech, kalendářích, grafech, jaké hodnoty budou zobrazeny v rozbalovacích seznamech v editačním formuláři apod.
- Odkaz na databázový dotaz je vždy součástí dialogu pro nastavení konkrétního ovládacího prvku, nejčastěji používaný s názvem „Zdroj dat“ nebo „Z dotazu“. Kliknutím na odkaz se zobrazí grafický návrhář databázových dotazů – „QueryBuilder“.
- Databázový dotaz je definován:
 - Databázovou tabulkou, ze které se záznamy načítají
 - Volbou, zda se jedná o běžný databázový dotaz, nebo agregační dotaz, který používá grupování
 - Sloupcem „Setřídít podle“, podle kterého jsou databázové záznamy seřazeny, včetně způsobu řazení – vzestupně (ASC) nebo sestupně (DESC)
 - Volitelně druhým sloupcem, podle kterého jsou databázové záznamy seřazeny, včetně způsobu řazení – vzestupně (ASC) nebo sestupně (DESC)
 - Sloupcem „Barva podle“, který určuje, zda a podle jakého sloupce se bude v nahlížecí tabulce zobrazovat u každého záznamu barevný obdélníček
 - Sloupcem „Časový úsek podle“, který určuje, zda a podle jakého sloupce se bude nad nahlížecí tabulkou zobrazovat filtr pro výběr časového úseku „od-do“.
 - Seznamem omezujících podmínek
 - Seznamem přijoinovaných databázových tabulek
 - V případě agregačního dotazu je dále definován:
 - Seznamem sloupců výsledné agregační tabulky
 - Seznamem hlaviček, které se zobrazují nad sloupci v nahlížecí tabulce
 - V případě grafu je dále definován seznamem barev, které budou použity na vykreslení jednotlivých sloupců grafu
 - Výběrem sloupce, podle kterého budou vyhodnoceny duplicitní řádky v načtené množině dat, a tyto řádky budou následně z této množiny odstraněny
- Výsledný SQL dotaz složený z klauzulí „SELECT“, „FROM“, „JOIN“, „WHERE“, „ORDER BY“ a „GROUP BY“ je vždy sestaven automaticky podle parametrů dotazu nastavených v grafickém návrhář databázových dotazů.
- Výsledkem databázového dotazu je vždy načtená množina dat z databáze, dočasně uložená v objektu typu „DataTable“. Tato množina dat je následně předána jednotlivým ovládacím prvkům za účelem jejího vyhodnocení nebo vizualizace. Před samotným předáním objektu „DataTable“ ovládacímu prvku může být spuštěna externí funkce, která má možnost měnit vlastnosti tohoto objektu – přidávat řádky, měnit hodnoty v jednotlivých sloupcích, případně řádky odstraňovat.
- Detailní popis návrháře databázových dotazů je uveden v samostatné příručce „Návrhář databázových dotazů“.

12.1 Běžný dotaz

- Běžný dotaz se používá u všech ovládacích prvků, které jako zdroj dat používají data načtená z databáze.
- Běžný dotaz je výchozí nastavení v návrhář databázových dotazů, a pro sestavení výsledného dotazu v jazyce SQL používá pouze klauzule „SELECT“, „FROM“, „JOIN“, „WHERE“ a „ORDER BY“. Nepoužívá tedy klauzuli „GROUP BY“.

12.2 Agregací dotaz

- Agregací dotaz se používá u nahlížecích tabulek a grafů, které zobrazují agregovaná data, načtená z databáze s pomocí klauzule „GROUP BY“.
- Agregací dotaz je definován v návrháři databázových dotazů:
 - zaškrtnutím pole „Agregací dotaz“
 - seznamem sloupců dotazu, podle kterých jsou záznamy seskupovány – těch, které se objeví ve výsledném SQL dotazu v klauzuli „GROUP BY“. Mezi podporované druhy seskupování patří:
 - Seskupit podle – běžný „GROUP BY ng_“
 - Seskupit podle hodiny ve dni
 - Seskupit podle hodiny
 - Seskupit podle dne
 - Seskupit podle týdne
 - Seskupit podle měsíce
 - Seskupit podle kvartálu
 - Seskupit podle roku
 - seznamem agregovaných sloupců – těch, které se objeví ve výsledném SQL dotazu ve spojení s funkcemi „COUNT“, „SUM“, „AVG“, „MAX“ a „MIN“, s možností nastavení podmínky pomocí klauzulí „CASE WHEN“. Mezi podporované druhy agregovaných sloupců patří:
 - Počet – „COUNT(*)“
 - Suma – „SUM(ng_)“
 - Průměr – „AVG(ng_)“
 - MAX – „MAX(ng_)“
 - MIN – „MIN(ng_)“
 - seznamem informativních sloupců – těch, které se objeví ve výsledném SQL dotazu jako textové nebo číselné konstanty, s možností nastavení podmínky pomocí klauzulí „CASE WHEN“, následně vyhodnocené lexikálním analyzátozem. Mezi podporované druhy informativních sloupců patří:
 - Text – „MIN(ng_)“
 - Výraz, formát dle – „MIN(ng_)“, následně vyhodnocený lexikálním analyzátozem podle matematického výrazu definovaného jako součást názvu sloupce
 - seznamem sloupců, podle kterých bude možné filtrovat v nahlížecí tabulce nebo grafu
 - seznamem hlaviček, do kterých budou seskupeny sloupce nahlížecí tabulky
 - seznamem barev, které budou použity v grafu na vykreslení jednotlivých sloupců grafu
- Jednotlivé sloupce agregací dotazu používají následující syntaxi pro nazývání sloupců:
„název@matematický_výraz@zobrazovaná_hodnota@podmínka“.
 - Název určuje název sloupce zobrazovaný v nahlížecí tabulce nebo grafu
 - Matematický výraz je volitelný údaj relevantní pro sloupce typu „Výraz, formát dle“, a definuje matematický vzorec použitý pro výpočet hodnoty zobrazované v nahlížecí tabulce nebo grafu

- Zobrazovaná hodnota je volitelný údaj relevantní pro sloupce typu „Výraz, formát dle“, a umožňuje zakrýt vypočtenou hodnotu konkrétním textovým řetězcem, případně skrýt celý sloupec v nahlížecí tabulce pomocí názvu sloupce zadaného jako „@@“.
- Podmínka je volitelný údaj relevantní pro sloupce typu „Výraz, formát dle“, a umožňuje zobrazit pouze řádky výsledné množiny dat, které vyhovují zadané podmínce.
- Příklady názvů sloupců:
 - „Součet prvního a druhého sloupce@#c0#+#c1#“
 - „Součet prvního a druhého sloupce@#c0#+#c1#@Skrytá hodnota“
 - „Součet s hodnotou na dalším řádku@+1“
 - „Součet s hodnotou na předchozím řádku@-1“
 - „Součet prvního a druhého sloupce a hodnoty na dalším řádku@+1#c0# + #c1#“
 - „Součet hodnot větších než 5@@@>5“
 - Zobrazení součtu sloupce v případě, kdy se suma sloupce defaultně nezobrazuje – „Sloupec@+sum“
 - Skrytí součtu sloupce, který se defaultně zobrazuje se sumou sloupce – „Sloupec@-sum“
- Jednotlivé hlavičky agregačního dotazu používají následující syntaxi pro nazývání hlaviček: „počet_sloučených_buněk@název_hlavičky“.
- Příklady názvů hlaviček:
 - 1. řádek: „5@X;1@Y“
 - 2. řádek: „2@A;3@B;1@C“

X		Y			
A		B			C
Sloupec 1	Sloupec 2	Sloupec 3	Sloupec 4	Sloupec 5	Sloupec 6
...

12.3 Vnořené záznamy

- Vnořený záznam je záznam přidružený ke svému rodičovskému záznamu pomocí cizích klíčů „pid“ a „pform“.
- Cizí klíče „pid“ a „pform“ jsou součástí každé databázové tabulky, a u každého databázového záznamu definují ID rodičovského záznamu (parent ID) a ID editačního formuláře (parent form), ve kterém se rodičovský záznam nachází.
- Cizí klíče „pid“ a „pform“ se nejčastěji odkazují na záznamy v jiné databázové tabulce – například položka zakázky se odkazuje na ID zakázky.
- Cizí klíče „pid“ a „pform“ se mohou odkazovat i na záznamy ve stejné databázové tabulce. Této vlastnosti využívá ovládací prvek „Tree“ – stromová struktura – který provádí vizualizaci databázových záznamů na základě rekurzivní metody.
- Používání cizích klíčů „pid“ a „pform“ je zcela automatické. NET Genium automaticky vyplňuje hodnotu těchto dvou sloupců při vytváření databázových záznamů v editačním formuláři podle toho, odkud záznam vznikl – z jakého záznamu byl uživatel do cílového editačního formuláře přesměrován za účelem vytvoření nového záznamu v databázi.

- Je-li uživatel do editačního formuláře přesměrován z nahlížecké stránky, nově uložený záznam bude mít hodnotu „pid“ a „pform“ nastavenou na „0“.
- Je-li uživatel do editačního formuláře přesměrován z již existujícího záznamu v databázi, nově uložený záznam bude mít hodnotu „pid“ a „pform“ nastavenou podle ID záznamu a editačního formuláře tohoto rodičovského záznamu.
- Používání cizího klíče „pform“ je z pohledu správce aplikace zcela minimální. Jen v ojedinělých případech se využívá hodnoty sloupce „pform“ – když jedna databázová tabulka s vnořenými záznamy slouží dvěma nebo více rodičovským tabulkám.
- Cizí klíče „pid“ ani „pform“ nemají ve výchozím nastavení zapnuté indexování. Kdykoliv se správce aplikace rozhodne tyto cizí klíče používat, je nezbytné zapnout indexování těchto sloupců přímo v nastavení editačního formuláře.
- Nahlížecké tabulky v editačních formulářích obsahují zaškrťovací pole „Zobrazit pouze vnořené záznamy“, které usnadňuje používání vnořených záznamů, a do definice databázového dotazu automaticky připojí omezující podmínku „PID = ID záznamu ve formuláři“.
- Nahlížecké tabulky na nahlížeckých stránkách obsahují zaškrťovací pole „Skrýt vnořené záznamy“, které usnadňuje používání vnořených záznamů, a do definice databázového dotazu automaticky připojí omezující podmínku „PID = 0“.

12.4 Omezující podmínky dotazu

- Databázový dotaz může obsahovat libovolné množství omezujících podmínek, které následují za klauzulí „WHERE“. Jednotlivé podmínky jsou mezi sebou oddělené pomocí operátorů „AND“ a „OR“, včetně možnosti využití levých a pravých závorek.
- Výsledný tvar podmínky dotazu v jazyce SQL je sestaven automaticky podle seznamu podmínek nastavených v grafickém návrháři. Detailní popis návrháře databázových dotazů je uveden v samostatné příručce „Návrhář databázových dotazů“.
- V podmínce dotazu je možné používat následující rozšiřující funkce:
 - Datový typ „Integer“ a „Long“ umožňuje zápis ve tvaru „Levá strana podmínky rovná se 1;2;3“. Výsledný tvar podmínky pak bude interpretován jako „Levá strana podmínky IN (1;2;3)“.
 - Datový typ „String“ umožňuje zápis ve tvaru „Levá strana podmínky rovná se (array)A;B;C“. Výsledný tvar podmínky pak bude interpretován jako „Levá strana podmínky IN ('A','B','C')“.
 - Levá strana podmínky umožňuje výběr ovládacího prvku „MultiListBox“ (hodnoty oddělené tabulátorem) spolu s textovou konstantou na pravé straně podmínky. Výsledný tvar podmínky pak bude interpretován jako „JoinText2(Levá strana podmínky, Pravá strana podmínky)“ u databáze Firebird nebo „dbo.JoinNtext(Levá strana podmínky, Pravá strana podmínky)“ u databáze MSSQL. SQL dotaz následně vyhodnotí všechny záznamy, které mají ve sloupci „MultiListBox“ na levé straně podmínky uloženou alespoň jednu hodnotu, která se rovná textové konstantě na pravé straně podmínky.
 - Levá strana podmínky umožňuje výběr ovládacího prvku „MultiListBox“ (hodnoty oddělené tabulátorem) spolu s ovládacím prvkem „MultiListBox“ na pravé straně podmínky. Výsledný tvar podmínky pak bude interpretován jako „JoinText2(Levá strana podmínky, Pravá strana podmínky)“ u databáze Firebird nebo „dbo.JoinNtext(Levá strana podmínky, Pravá strana podmínky)“ u databáze MSSQL. SQL dotaz následně vyhodnotí všechny záznamy, které mají ve sloupci „MultiListBox“ na levé straně podmínky uloženou alespoň jednu hodnotu, která je uložena ve sloupci „MultiListBox“ na pravé straně podmínky.
 - Datový typ „String“ umožňuje zápis ve tvaru „Levá strana podmínky rovná se (mlb)A#tab#B#tab#C“. Výsledný tvar podmínky pak bude interpretován jako „JoinText2(Levá strana podmínky, 'A B C')“ u databáze Firebird nebo „dbo.JoinNtext(Levá strana podmínky, 'A B C')“ u databáze MSSQL. SQL dotaz následně vyhodnotí všechny záznamy, které mají ve sloupci s hodnotami oddělenými tabulátorem na levé straně podmínky uloženou alespoň jednu hodnotu, která je součástí textové konstanty na pravé straně podmínky.

- Podmínka dotazu může být definována také pomocí SQL syntaxe, která obsahuje jak levou, tak pravou stranu podmínky. Levá strana podmínky včetně operátoru může být libovolná, pravá strana podmínky musí začínat slovem „OK#crLf#“, a za ní musí následovat znění podmínky v jazyce SQL, například:
 - OK#crLf#0=0
 - OK#crLf#0=1
 - OK#crLf#EQUALS(ng_tb, "", 0=0, ng_tb = FORMATSTRINGSQL(#ng_tb#))
 - OK#crLf#EQUALS(ng_tb, "", 0=1, ng_tb = FORMATSTRINGSQL(#ng_tb#))

12.5 Připojení (joiny)

- Databázový dotaz může obsahovat libovolné množství připojení, která jsou sestavená pomocí klauzule „JOIN“.
- Výsledný tvar připojení v jazyce SQL je sestaven automaticky podle seznamu připojení nastavených v grafickém návrháři. Detailní popis návrháře databázových dotazů je uveden v samostatné příručce „Návrhář databázových dotazů“.
- Připojení může být definováno také pomocí SQL syntaxe. Levá strana podmínky připojení může být libovolná, pravá strana podmínky připojení musí začínat slovem „OK#crLf#“, a za ní musí následovat znění připojení pomocí jazyka SQL, například:
 - OK#crLf#INNER JOIN ng_abc J1 ON J1.pid = ng_formular.id
 - Definice připojení musí respektovat používané aliasy, takže první připojení se musí jmenovat J1, druhé J2 atd.
 - Definice připojení musí respektovat název zdrojové databázové tabulky.

13 Proměnné a identifikátory

- Proměnná je textový identifikátor, který začíná a končí znakem „#“, a uchovává v sobě informaci o webové aplikaci, databázi, aktuálně přihlášeném uživateli, editačních formulářích, nahlížecích stránkách, nebo ovládacích prvcích obecně.
- Identifikátory jsou na straně serveru nahrazovány za konkrétní hodnoty vyjádřené textovým řetězcem.
- Mezi proměnné patří
 - Obecné proměnné – například „#loginname#“ nebo „#today#“
 - Identifikátory databázových ovládacích prvků
 - Aktuální hodnota zadaná uživatelem v editačním formuláři – „#ng_#“
 - Aktuální hodnota zadaná uživatelem v editačním formuláři, dotazovaná z javascriptového kódu – „#ng_:value#“
 - Původní historická hodnota, resp. aktuální hodnota v databázi – „#ng_:history#“
 - Hodnota ovládacího prvku „MultiListBox“, kde jsou jednotlivé textové položky převedené na jejich „id“ v databázi – „#ng_:ids#“
 - Javascriptové proměnné – například „nID“
 - Skriptové proměnné – například „#A#“
 - Proměnné používané v tiskových šablonách – například „#ng_#“, „#c0“ nebo „#remove#“
- Proměnné mohou používat pouze administrátoři při tvorbě aplikací. Běžný uživatel nemá z prostředí aplikace přístup k hodnotám proměnných, resp. proměnné používá pouze zprostředkovaně pomocí aplikací navržených administrátory.
- Identifikátory databázových ovládacích prvků v editačním formuláři se chovají jinak uvnitř javascriptového kódu.
 - Použití proměnné „#ng_#“ slouží pro přístup k objektu databázového ovládacího prvku uvnitř objektového modelu HTML dokumentu editačního formuláře – proměnná je v takovém případě nahrazena za „document.getElementById()“ nebo „document.forms[0].“ v závislosti na použitém webovém prohlížeči.
 - K samotné hodnotě databázového ovládacího prvku momentálně uložené v databázi se z javascriptového kódu přistupuje pomocí proměnné „#ng_:value#“.
 - Pokud uživatel nemá právo na ovládací prvek spojený s danou proměnnou, dojde k nahrazení proměnné za hodnotu uloženou v tomto prvku v databázi. Může tedy dojít i k nahrazení proměnné za prázdnou hodnotu, což může v javascriptovém kódu způsobit syntaktickou chybu, například „var o = ;“.
- Detailní popis proměnných je uveden v samostatné příručce „Proměnné“.

14 Serverové funkce

- Serverové funkce slouží k volání programového kódu na straně serveru, který vrací hodnotu vyjádřenou textovým řetězcem.
- Serverové funkce používají parametry uzavřené v kulatých závorkách, a oddělené čárkou. Parametry mohou být umístěny ve dvojitéch uvozovkách například v situaci, kdy jeden z parametrů obsahuje znak dělicí čárky, nebo začíná či končí mezerou. Funkce, které nemají žádné parametry, i tak při jejich volání musí obsahovat kulaté závorky.
- Serverové funkce mohou používat pouze administrátoři při tvorbě aplikací. Běžný uživatel nemá z prostředí aplikace přístup k volání serverových funkcí, resp. serverové funkce volá pouze zprostředkovaně pomocí aplikací navržených administrátory.
- Detailní popis serverových funkcí je uveden v samostatné příručce „Serverové funkce“.

15 Externí funkce

- Externí funkce slouží k volání vlastního programového kódu v jazyce C# na straně serveru, který vrací hodnotu vyjádřenou textovým řetězcem.
- Pro volání externích funkcí slouží serverová funkce „ngef(string id, string arg0, string arg1, string arg2, ...)“, která obsahuje vyžadovaný parametr „id“ s identifikátorem externí funkce, a dále libovolné množství dalších volitelných parametrů, které jsou do externí funkce předány jako „string[] args“.
- Externí funkce mohou používat pouze administrátoři při tvorbě aplikací. Běžný uživatel nemá z prostředí aplikace přístup k volání externích funkcí, resp. externí funkce volá pouze zprostředkovaně pomocí aplikací navržených administrátory.
- Detailní popis programování externích funkcí je uveden v samostatné příručce „Externí funkce“.

16 Skript

16.1 Základní informace

- Skript je programový kód, který se spouští na straně serveru
 - v serverových událostech editačních formulářů,
 - **OnBeforeOpen** – skript, který se provede před otevřením editačního formuláře po kliknutí na tlačítko „Uložit a nový“
 - **OnAfterOpen** – skript, který se provede po otevření editačního formuláře
 - **OnBeforeSave** – skript, který se provede před uložením záznamu do databáze
 - **OnAfterSave** – skript, který se provede po uložení záznamu do databáze
 - **OnBeforeDelete** – skript, který se provede před smazáním záznamu z databáze
 - **OnAfterDelete** – skript, který se provede po smazání záznamu z databáze
 - po stisknutí tlačítka umístěného v editačním formuláři,
 - po stisknutí tlačítka umístěného na nahlížecí stránce nebo
 - při pokusu o vytvoření nového záznamu z ovládacího prvku „TimeTable“ a „Planner“.
- Odkaz na skript je vždy součástí dialogu pro nastavení konkrétního ovládacího prvku – nejčastěji „ActionButtons“ v případě serverových událostí editačních formulářů, nebo „Button“ v případě tlačítek umístěných v editačních formulářích nebo na nahlížecích stránkách – používaný s názvem „Upravit skript (počet řádek skriptu)“. Kliknutím na odkaz se zobrazí grafický návrhář skriptů.
- Skript se skládá z jednotlivých příkazů, které jsou ve formě číslovaných řádek skriptu vkládány pod sebe. Každá řádka skriptu obsahuje instrukci, jaký příkaz se má vykonat. NET Genium prochází řádku po řádce, a v závislosti na instrukcích vykonává příslušné příkazy.
- Základní skriptové příkazy tvoří:
 - Deklarace proměnné
 - Příkaz nastaví datový typ skriptové proměnné.
 - Obsahuje-li proměnná nějakou hodnotu, nebo je-li proměnná již dříve deklarována, dojde ke změně datového typu proměnné a nastavení její hodnoty na databázovou hodnotu „null“.
 - Přiřazení hodnoty z dotazu do proměnné nebo do ovládacího prvku ve formuláři
 - Příkaz načte hodnotu z databázového dotazu, a uloží ji buď do skriptové proměnné nebo do databázového ovládacího prvku, jde-li o skript uvnitř editačního formuláře.
 - Obsahuje-li proměnná nějakou hodnotu, nebo je-li proměnná již dříve deklarována, dojde ke změně datového typu proměnné a k přepsání její hodnoty.
 - Uložení proměnné zpět do databáze
 - Příkaz uloží pole hodnot proměnné zpět do databáze na místo, odkud bylo načteno.
 - Proměnná nesmí v průběhu skriptu změnit počet svých hodnot, např. seskupením pomocí funkce „GROUP(##)“.
 - Přiřazení více hodnot z dotazu do proměnných nebo do ovládacích prvků ve formuláři

- Příklad načte hodnoty více sloupců z databázového dotazu, a uloží je buď do skriptové proměnné nebo do databázových ovládacích prvků, jde-li o skript uvnitř editačního formuláře.
- Cyklus
 - Příklad provede cyklus na konci ohraničený příkazem „Další...“ tolikrát, kolik má zvolená řídicí proměnná prvků. Obsahuje-li proměnná 0 prvků, cyklus neproběhne ani jednou, a skript pokračuje za příkazem „Další...“.
 - Význam cyklu spočítá v procházení jednotlivých prvků (i duplicit) v poli hodnot řídicí proměnné. V každé iteraci cyklu se řídicí proměnná chová jako jednoprvkové pole, resp. jako jedna hodnota. Po skončení cyklu se stává opět původní proměnnou obsahující pole hodnot.
 - Obsahuje-li proměnná v určité iteraci cyklu databázovou hodnotu „null“ (hodnota není definovaná), přeskočí cyklus na další možnou iteraci, kdy je hodnota proměnné již definovaná.
 - Je-li řídicí proměnná cyklu načtena spolu s dalšími proměnnými příkazem „Přiřadit více hodnot z dotazu“, chovají se během procházení cyklu tyto proměnné stejně jako řídicí proměnná. Například při současném načtení proměnných „A“ a „B“ je jedno, která z těchto dvou proměnných bude použita jako řídicí. Zvolená řídicí proměnná by pouze neměla obsahovat prvky s databázovou hodnotou „null“, proto se doporučuje jako řídicí proměnnou používat primární klíč „id“.
- Přiřazení hodnoty do proměnné nebo do ovládacího prvku ve formuláři
 - Příklad přiřadí hodnotu do skriptové proměnné nebo do databázového ovládacího prvku. Příklad umožňuje přiřazení jedné hodnoty, nebo matematického výrazu – součtu, rozdílu, násobku nebo podílu dvou hodnot.
 - Je-li proměnná polem hodnot – například u příkazu „[A ▼] = [A ▼][+ ▼][1]“ – početní operace proběhne u všech jejích prvků.
- Porovnání hodnot dvou proměnných mezi sebou či porovnání hodnoty zadané uživatelem v editačním formuláři s proměnnou
- Skok na řádek
- Vyvolání přerušování
- Odeslání e-mailu
- Smazání záznamu z databáze
 - Příklad smaže všechny záznamy, které vyhovují podmínkám databázového dotazu.
- Přesměrování uživatele do editačního formuláře za účelem vytvoření nového záznamu
 - Příklad naplánuje přesměrování uživatele do aktuálně otevřeného editačního formuláře za účelem vytvoření nového záznamu. K samotnému přesměrování dojde až do úspěšném dokončení celého skriptu.
 - Příklad je přístupný pouze ve skriptu „OnAfterSave“, „OnBeforeOpen“ nebo „OnAfterDelete“.
- Vytvoření nového záznamu v databázi
 - Příklad vytvoří jeden nebo několik nových záznamů v libovolné databázové tabulce. Pro vytváření několika záznamů najednou se používá databázový dotaz. Ten říká, kolik záznamů bude vytvořeno, a při vytváření dává k dispozici načtené hodnoty ze zdrojové databázové tabulky. Prochází tedy záznam po záznamu a z takto získaných hodnot vytváří nové záznamy.
 - Poslední záznam vytvořený tímto příkazem ve skriptu „OnBeforeSave“ má své unikátní „id“ záznamu uložené v proměnné „#nid#“, které je možné použít v následném skriptu „OnAfterSave“.
- Uložení hodnoty do databáze

- Příkaz zapíše zvolenou hodnotu na libovolné místo v databázi definované databázovým dotazem. Je-li výsledkem databázového dotazu více záznamů, bude hodnota uložena do všech těchto záznamů v databázi.
- Uložení více hodnot do databáze
 - Příkaz zapíše zvolené hodnoty na libovolné místo v databázi definované databázovým dotazem. Je-li výsledkem databázového dotazu více záznamů, budou hodnoty uloženy do všech těchto záznamů v databázi.
- Komentář
 - Příkaz slouží pro interní poznámky autora skriptu, nebo pro volání externí funkcí.
- Ve skriptu je možné používat:
 - Skriptové proměnné, které se používají na ukládání dočasných hodnot – například „#A#“
 - Textové proměnné, které se pomocí identifikátorů odkazují na hodnoty ovládacích prvků zadané uživatelem v editačním formuláři – například „#ng_#“
 - Textové proměnné, které se pomocí identifikátorů odkazují na aktuální hodnoty ovládacích prvků v databázi – například „#ng_:history#“
 - Volání serverových funkcí NET Genia – například „SUBSTRING(#A#, 0, 2)“
 - Volání externích funkcí napsaných v jazyce C# – například „ngef(MyFirstFunction, #A#)“
- Skript je programovací jazyk, který má vlastní syntaxi, avšak nepodporuje programové bloky. Jeho činnost je zajišťována pouze sadou výše zmíněných základních příkazů vkládaných do návrháře skriptů. Programové bloky nebo složitější algoritmy, které není možné obsloužit pomocí základních příkazů ve spojení s proměnnými a serverovými funkcemi, se musí naprogramovat pomocí externích funkcí napsaných v jazyce C#.
- Skript přiřazený tlačítku na nahlížecí stránce je možné spustit i externě přímo z operačního systému MS Windows pomocí programu „RunScript.exe“. Program „RunScript.exe“ je umístěn na disku serveru v adresáři „NETGenium\bin“.
- Během provádění skriptu může dojít k chybám či přerušením, a to buď na základě syntaktické chyby ve skriptu, chyby při konverzích hodnot, při ukládání dat do databáze, nebo při programovém vyvolání přerušení skriptu. Ve všech případech to znamená ukončení provádění skriptu, návrat zpět do editačního formuláře nebo na nahlížecí stránku, odkud byl skript vyvolán, a zobrazení chybového hlášení uživateli. U skriptů „OnBeforeSave“ a „OnBeforeDelete“ takové přerušení znamená zamezení plánovaného uložení, resp. smazání záznamu.
- Detailní popis návrháře skriptů je uveden v samostatné příručce „Návrhář skriptů“.

16.2 Nový záznam vs existující záznam v databázi

- Při vyhodnocování skriptu je často nutné rozlišit, zda má uživatel otevřený editační formulář s novým dosud neuloženým záznamem, nebo zda edituje již existující záznam v databázi. Pro tyto účely slouží primární klíč „id“, resp. proměnná „#id#“, která má v případě dosud neuloženého záznamu hodnotu „0“. K vygenerování nového „id“ dochází až po úspěšném uložení záznamu pomocí skriptu „OnBeforeSave“. K „id“ nově vytvořeného záznamu je možné přistupovat ve skriptu „OnAfterSave“, kde je toto „id“ již známé.

16.3 Skriptové proměnné

- Při práci se skriptovými proměnnými je nutné danou proměnnou vždy nejdříve nadeklarovat – definovat datový typ proměnné. Mezi podporované datové typy patří:
 - String – textový řetězec

- Integer – celé číslo
- Long – velké celé číslo
- Double – desetinné číslo
- Date – datum
- Deklaraci proměnné lze provést následujícími způsoby:
 - Pomocí skriptového příkazu na deklaraci proměnné
 - Pomocí skriptového příkazu na načtení hodnoty z dotazu – datový typ proměnné se automaticky převezme z typu databázového sloupce, ze kterého je hodnota do proměnné načtena
 - Pomocí příkazu „String(abc)“, „Integer(1)“, „Long(1)“, „Double(1)“ nebo „Date(#today#)“
- Proměnné, které jsou načtené z dotazu, se automaticky stávají polem hodnot. Tyto hodnoty je možné procházet pomocí skriptového příkazu „Cyklus“ – interpretace příkazu „foreach (var item in items)“ používaného v jazyce C# – nebo vyhodnocovat pomocí funkcí „AVERAGE(#A#)“, „COUNT(#A#)“, „FIRST(#A#)“, „GROUP(#A#)“, „LAST(#A#)“ a „SUM(#A#)“. Je-li výsledkem dotazu 0 záznamů, jde o pole s nulovým počtem prvků.
- Pole hodnot je možné deklarovat i ručně pomocí následujících serverových funkcí:
 - „StringArray(seznam hodnot oddělených středníkem)“
 - „IntegerArray(počet hodnot)“
 - „LongArray(počet hodnot)“
 - „DoubleArray(seznam hodnot oddělených středníkem)“
 - „DateArray(seznam hodnot oddělených středníkem)“
- I ručně generované proměnné typu pole hodnot je možné procházet pomocí skriptového příkazu „Cyklus“.
- Proměnné používají databázovou hodnotu „null“. Při základní deklaraci proměnné, která není polem hodnot, je její hodnota vždy nastavena na databázovou hodnotu „null“. Kdykoliv později je možné do proměnné uložit databázovou hodnotu „null“ buď pomocí proměnné „#null#“, nebo přiřazením jiné proměnné, která nese databázovou hodnotu „null“ – například byla právě deklarovaná, a dosud do ní nebyla uložena hodnota. S proměnnými je možné pracovat podobně jako s hodnotami databázových sloupců, a testovat je pomocí operátorů „je definováno“ a „není definováno“.

16.4 Uložení hodnoty proměnné do databázového ovládacího prvku

- Hodnotu proměnné lze přiřadit do databázového ovládacího prvku, např. po vyplnění editačního formuláře uživatelem, a následném spuštění skriptu pomocí tlačítka. Je-li proměnná polem hodnot, je do ovládacího prvku přiřazena pouze jeho první hodnota v pořadí. Není-li proměnná definovaná či její pole obsahuje 0 prvků, je do ovládacího prvku přiřazena databázová hodnota „null“.

16.5 Porovnávání hodnot proměnných

- Ve skriptu je možné porovnávat hodnoty dvou proměnných. Je-li jedna z proměnných polem hodnot, jsou porovnávány pouze první hodnoty těchto polí. Dále je možné ověřit, obsahuje-li pole hodnot určitou hodnotu. Pro tento test se používá operátor „obsahuje“. Je však důležité nezaměňovat funkci tohoto operátoru s použitím testu u běžné proměnné deklarované ručně a nesoucí textový řetězec. Tato proměnná není výsledkem dotazu, a tudíž i není polem hodnot. Operátor „obsahuje“ pak slouží k ověření, obsahuje-li řetězec konkrétní podřetězec.

16.6 Konverze hodnot na jiný datový typ

- Zápis proměnné v zadávacím poli příkazu skriptu může sloužit pro konverzi hodnoty proměnné na požadovaný datový typ.
- Příklad konverze číselné hodnoty na textový řetězec:
 - [A ▼] = [String ▼]
[B ▼] = [Integer ▼]
[B ▼] = [2]
[A ▼] = [#B#]
- Příklad konverze textového řetězce na číselnou hodnotu:
 - [A ▼] = [Integer ▼]
[B ▼] = [String ▼]
[B ▼] = [2]
[A ▼] = [#B#]

17 Javascript

- Javascript je programový kód, který se spouští na straně klienta ve webovém prohlížeči
 - z ovládacího prvku „JavaScript“ v editačním formuláři,
 - z ovládacího prvku „JavaScript“ na nahlížecí stránce,
 - po stisknutí tlačítka umístěného v editačním formuláři,
 - po stisknutí tlačítka umístěného na nahlížecí stránce nebo
 - při vyvolání javascriptové události databázových ovládacích prvků v editačním formuláři – „OnBlur“, „OnChange“, „OnClick“ a „OnFocus“.
- V javascriptu je možné používat:
 - Textové proměnné NET Genia – například „#today#“
 - Textové proměnné, které se pomocí identifikátorů odkazují na objekty databázových ovládacích prvků editačního formuláře uvnitř objektového modelu HTML dokumentu – „#ng_#“
 - Textové proměnné, které se pomocí identifikátorů odkazují na hodnoty ovládacích prvků zadané uživatelem v editačním formuláři – „#ng_:value#“
 - Textové proměnné, které se pomocí identifikátorů odkazují na aktuální hodnoty ovládacích prvků v databázi – „#ng_:history#“
 - Rozšířené textové proměnné, které se pomocí identifikátorů odkazují na tabulku s ovládacím prvkem v editačním formuláři – „#ng_:Table#“
 - Ovládací prvky jsou v editačním formuláři řazeny pod sebou, a jsou umístěny v tabulkách – `<table>...</table>`.
 - Každý databázový ovládací prvek je umístěn v samostatné tabulce, kterou je možné skrývat nebo zobrazovat – například `#ng_:Table#.style.display = 'none'`;
 - Rozšířené textové proměnné, které se pomocí identifikátorů odkazují na label s názvem ovládacího prvku v editačním formuláři – „#ng_:Name#“
 - Názvy ovládacích prvků jsou v editačním formuláři zobrazovány nalevo vedle samotného ovládacího prvku, a jsou umístěny v labelech – `<label>...</label>`.
 - Každý databázový ovládací prvek má svůj vlastní label, kterému je možné měnit zobrazovanou hodnotu – například `#ng_:Name#.innerHTML = 'abc'`;
 - Volání javascriptových funkcí NET Genia – například „control_Enable(#ng_#);“
 - Volání serverových funkcí NET Genia – například „SQL(SELECT ng_ FROM ng_)“ nebo „SQLARRAY(SELECT * FROM ng_)“
 - Volání externích funkcí napsaných v jazyce C# – například „ngef(MyFirstFunction, #today#)“
- Javascript se nejčastěji používá na
 - přečtení nebo nastavení hodnoty databázových ovládacích prvků,
 - validaci hodnot zadaných uživatelem do databázových ovládacích prvků,
 - změnu vlastností nebo chování databázových ovládacích prvků,
 - skrývání nebo zobrazování částí editačního formuláře,
 - načítání nebo ukládání hodnot z databáze pomocí ajaxu.

- Detailní popis javascriptových funkcí je uveden v samostatné příručce „Javascriptové funkce“.

18 Tisk

- Tisk do tiskové šablony umožňují tlačítka umístěná v editačním formuláři nebo na nahlížecí stránce. Mezi podporované tiskové šablony patří soubory „xls“, „xlsx“, „doc“, „docx“, „html“, „txt“, „csv“ a „pdf“. Tiskové šablony „xlsx“ a „docx“ podporují automatickou konverzi do „pdf“.
- Tiskové šablony musí být uloženy v adresáři „NETGenium\Templates“. Do tohoto adresáře je možné soubory nakopírovat napřímo z plochy serveru, nebo prostřednictvím dialogu pro nastavení NET Genia na záložce „Tiskové šablony“. Z těchto souborů je následně možné vybírat přímo ve vlastnostech tlačítka zvolenou tiskovou šablonu.
- V tiskových šablonách mohou být používány pouze identifikátory databázových ovládacích prvků a skriptové proměnné. Skriptové proměnné je možné naplnit před samotným tiskem pomocí skriptu.
- Na místo v šabloně, kam je potřeba dosadit hodnoty, se vkládají identifikátory databázových ovládacích prvků nebo názvy proměnných. NET Genium otevře tiskovou šablonu, vyhledá výskyty všech identifikátorů, a nahradí je příslušnými hodnotami.
 - Dosazení hodnot databázových ovládacích prvků do tiskové šablony z aktuálně otevřeného záznamu v editačním formuláři používá identifikátory ve zkráceném tvaru ve formátu „#ng_ovládací_prvek#“.
 - Tisk záznamů nahlížecí tabulky vyžaduje používání identifikátorů v plném tvaru buď ve formátu
 - „#ng_formulář#.#ng_ovládací_prvek#“ nebo
 - „#datagrid1#.#ng_ovládací_prvek#“, „#datagrid2#.#ng_ovládací_prvek#“, atd.
 - „#calendar1#.#ng_ovládací_prvek#“, „#calendar2#.#ng_ovládací_prvek#“, atd.
 - „#timetable1#.#ng_ovládací_prvek#“, „#timetable2#.#ng_ovládací_prvek#“, atd.
 - „#planner1#.#ng_ovládací_prvek#“, „#planner2#.#ng_ovládací_prvek#“, atd.
 - Druhý formát plného identifikátoru s pojmenováním typu vizuálního ovládacího prvku a s číslem jeho pořadí je doporučovaný, protože zohledňuje situace, kdy je v editačním formuláři nebo na nahlížecí stránce umístěno více nahlížecích tabulek se stejným zdrojem dat, jen s různými omezujícími podmínkami.
 - NET Genium rozpoznává výskyt identifikátorů v plném tvaru, a zajistí automatické vkládání hodnot jednotlivých záznamů pod sebe.
 - Jsou-li identifikátory umístěny vedle sebe v jedné řádce, je výstupem tisku tabulka, kde má každý záznam tabulky svůj řádek v tiskové šabloně.
 - Jsou-li identifikátory umístěny vedle sebe ve dvou řádcích, je výstupem tisku tabulka, kde se každý záznam tabulky skládá ze dvou řádek v tiskové šabloně.
 - Tisk z nahlížecí stránky, která obsahuje pouze jednu nahlížecí tabulku, nevyžaduje identifikátory v plném tvaru, stačí identifikátory ve zjednodušeném tvaru.
 - Tisk statistické nahlížecí tabulky s agregovanými sloupci používá identifikátory sloupců „c0“, „c1“, „c2“ atd., tj. podle jejich pořadí v hlavičce tabulky, s číslováním od nuly.
- Tiskové šablony vytvořené v aplikaci MS Word musí mít identifikátory umístěné v poli typu „MergeField“.
- Hodnoty mohou být v tiskové šabloně formátovány podle šablon pomocí syntaxe „#identifikátor@formát#“.
- Po úspěšném vytvoření tiskové sestavy se automaticky spouští externí funkce „NETGenium.Print“, ve které je možné ovlivnit jak název výsledného souboru, tak i obsah nebo formát souboru.
- Detailní popis programování tiskových šablon ve formátu „pdf“ je uveden v samostatné příručce „Programování tiskových šablon ve formátu PDF“.

19 Aplikace „Nastavení“

- NET Genium obsahuje výchozí sadu editačních formulářů a k nim náležících nahlížecích stránek, které jsou bezpodmínečně nutné pro správnou činnost celé aplikace. Tyto editační formuláře jsou umístěny v aplikaci „Nastavení“.
- Aplikaci „Nastavení“ není možné smazat, ani přesunout do jiné aplikační skupiny.
- Úpravy editačních formulářů a nahlížecích stránek v této aplikaci může provádět pouze jeden uživatel s nejvyšším oprávněním – uživatel „Administrator“ s databázovým ID 1.
- Aplikace „Nastavení“ obsahuje následující editační formuláře a k nim náležící nahlížecí stránky:
 - **Uživatel** – databázová tabulka „susers“ se seznamem uživatelů
 - Celé jméno – celé jméno uživatele
 - Příjmení – příjmení uživatele
 - Jméno – jméno uživatele
 - Titul – titul uživatele
 - Přihlašovací jméno – přihlašovací jméno uživatele
 - Heslo – heslo uživatele
 - Jazyk – jazykové nastavení uživatele (cs, de, en, fr, sk)
 - Účet systému Windows – název doménového účtu uživatele v Active Directory, používaný pro zajištění automatického přihlašování do NET Genia bez nutnosti zadávání přihlašovacího jména a hesla
 - E-mail – e-mailová adresa uživatele
 - M-Heslo – heslo pro účet na mobilním zařízení Android
 - Skupina oprávnění – cizí klíč do číselníku skupin oprávnění, který definuje seznam uživatelských skupin, ve kterých je uživatel členem. Přiřazení uživatele do skupiny oprávnění je preferovaný způsob přidělování práv, nicméně je možné i ruční přiřazení seznamu uživatelských skupin viz další ovládací prvek „Uživatelské skupiny“.
 - Uživatelské skupiny – seznam uživatelských skupin, ve kterých je uživatel členem. Ruční přiřazení uživatelských skupin každému uživateli zvlášť není preferovaný způsob přidělování práv, doporučovaný způsob je výběrem skupiny oprávnění viz předchozí ovládací prvek.
 - **Uživatelská skupina** – databázová tabulka „susergroups“ se seznamem uživatelských skupin
 - Uživatelská skupina – název uživatelské skupiny
 - Popis – popis uživatelské skupiny
 - **Skupina oprávnění** – databázová tabulka „srightsgroups“ se seznamem skupin oprávnění
 - Skupina oprávnění – název skupiny oprávnění
 - Popis – popis skupiny oprávnění
 - Uživatelské skupiny – seznam uživatelských skupin, které definují skupinu oprávnění
 - **Státní svátek** – databázová tabulka „sholiday“ se seznamem aktuálních státních svátků. Obsah této tabulky je generován automaticky podle jazykového nastavení NET Genia.
 - Název svátku – název státního svátku
 - Datum – datum státního svátku
 - Půldenní – zaškrtnuté pole pro účely volby půldenních státních svátků v Německu

- **Vzhled** – databázová tabulka „slayout“ se seznamem používaných grafických vzhledů NET Genia
 - Název – název vzhledu
 - Uložit jako aktuální vzhled – zaškrtnuté pole pro účely označení vzhledu jako aktuálního, a pro účely následného přegenerování CSS stylů.
 - Název fontu
 - Velikost textu
 - Barva písma
 - Záhlaví – Logo
 - Záhlaví – Obrázek na pozadí
 - ...
- **Neplatné přihlášení** – databázová tabulka „invalidlogins“ se seznamem neplatných přihlášení do NET Genia
 - Datum – datum neplatného přihlášení
 - IP adresa – IP adresa, ze které byl učiněn pokus o přihlášení
 - Přihlašovací jméno – použité přihlašovací jméno
- **Statistika** – databázová tabulka „statistics“ se statistikou využívání nahlížecích stránek NET Genia
 - Nahlížecí stránka – navštívená nahlížecí stránka
 - Uživatel – přihlašovací jméno uživatele, který stránku navštívil
 - Datum – datum návštěvy
- **Statistika 2** – databázová tabulka „statistics2“ se statistikou využívání nahlížecích stránek NET Genia
 - Nahlížecí stránka – navštívená nahlížecí stránka
 - Session – počet návštěv
- **Statistika 3** – databázová tabulka „statistics3“ se statistikou využívání nahlížecích stránek NET Genia
 - Nahlížecí stránka – navštívená nahlížecí stránka
 - Uživatel – přihlašovací jméno uživatele, který stránku navštívil
 - Session – počet návštěv
- **Synchronizace** – databázová tabulka „synchro“ se seznamem chyb vzniklých během synchronizace
 - Datum chyby
 - Editační formulář – název editačního formuláře, ve kterém došlo k chybě
 - ID záznamu – ID databázového záznamu, u kterého došlo k chybě
 - Chyba – popis chyby

20 Nastavení NET Genia

- Nastavení NET Genia se skládá z jedné části uložené v databázi, konfigurované pomocí webové aplikace v administrátorském režimu, a z konfiguračních souborů, které jsou uloženy na disku v adresáři „NETGenium\Config“.

20.1 Nastavení NET Genia konfigurované ve webové aplikaci

- Ikona pro vyvolání dialogu nastavení NET Genia je umístěna v toolbaru v navigační oblasti v administrátorském režimu. Toolbar se zobrazí kliknutím na ikonu ozubeného kolečka, nebo na nápis NET Genium umístěný napravo vedle této ikony.
- Detailní popis nastavení NET Genia je uveden v samostatné příručce „Nastavení NET Genia“.

20.2 Konfigurační soubory

- Konfigurační soubory obsahují údaje, které jsou vyžadované pro správný běh webové aplikace, ale nejsou součástí databáze. Nachází se v adresáři „NETGenium\Config“ ve formě textových souborů.
- **ActiveLayout.txt** – databázové ID vzhledu, který bude použitý jako výchozí, nezávisle na nastavení aktivního vzhledu v databázových záznamech nakonfigurovaných vzhledů NET Genia v aplikaci „Nastavení“
- **BlindFriendly.txt** – existence tohoto souboru zajistí, že nahlížeč tabulky nemají dynamicky zobrazovanou lištu s nástroji, ale statickou, která je stále zobrazena pod titulkem nahlížeč tabulky
- **ConnectionString.txt** – přípojovací řetězec k databázi NET Genia
- **Copyright.txt** – zástupný HTML kód pro copyright zobrazovaný v navigační oblasti. Pokud soubor neexistuje, zobrazuje se v navigační oblasti textový řetězec „© 20xx NetGenium“.
- **CustomIcons.txt** – seznam zástupných ikon, které nahrazují výchozí ikony používané v aplikaci. Na každé řádce souboru je umístěna definice jedné ikony v následujícím formátu:
„Images/Picker.gif:Images/Custom/Picker.svg“
- **DefaultLanguage.txt** – dvoupísmenná zkratka výchozího jazyka, ve kterém je lokalizována základní aplikace „Nastavení“, a který byl nastaven uživateli „Administrator“ a „Anonymous“ při jejich vytvoření („cs“, „de“, „en“, „fr“, „sk“)
- **Demo.txt** – existence tohoto souboru zajistí, že ve webové aplikaci nebude možné ukládání nebo mazání čehokoliv. Soubor může volitelně obsahovat seznam databázových tabulek oddělených enterem, ve kterých budou změny povoleny. Například „susers_vf“ zajistí, že bude možné uživatelské přizpůsobení nahlížeč tabulek.
- **DenyCookieless.txt** – existence tohoto souboru zajistí, že bude webová aplikace vyžadovat používání cookies na straně klienta. Webová aplikace následně
 - nebude používat vkládání SessionID do URL každé webové stránky
„(../netgenium/(S(33cipeb0js15bo1t1xhkupr4))/default.aspx)“ a
 - nebude umožňovat přihlášení v jednom webovém prohlížeči pod různými uživatelskými účty na různých záložkách – webový prohlížeč slučuje jednotlivé session.
- **DenyExport.txt** – existence tohoto souboru zajistí, že bude zakázaný export aplikací
- **DenyLogin.txt** – existence tohoto souboru zajistí, že bude zakázané přihlašování do aplikace z jakéhokoliv jiného místa než z plochy serveru, resp. z IP adresy „127.0.0.1“. Soubor může volitelně obsahovat URL, na kterou bude uživatel přesměrován při pokusu o přihlášení do aplikace.
- **DenySynchro.txt** – existence tohoto souboru zajistí, že bude zakázaná synchronizace dat

- **DenyTrimStream.txt** – existence tohoto souboru zajistí, že bude zakázané automatické zkracování HTML kódu všech stránek aplikace do minimální podoby – z HTML kódu budou odstraňována prázdná místa mezi koncem jednoho a začátkem druhého tagu.
- **Developer.txt** – existence tohoto souboru zajistí, že budou chyby vzniklé v aplikaci zobrazovány ve své nativní podobě, bez uživatelsky přívětivé podoby
- **EncryptionKey.txt** – šifrovací klíč pro šifrování sloupců „TextArea“ a „RichTextBox“. Při neexistenci tohoto souboru se pro šifrování dat používá výchozí šifrovací klíč.
- **ExecutionTimeout.txt** – maximální délka zpracování requestu, resp. webové stránky na straně serveru uváděná v hodinách, která se používá pro navýšení výchozí délky 5 hodin, typicky při potřebě spuštění dlouho trvajících skriptů. Maximální délka zpracování requestu má vliv na hodnotu atributu „executionTimeout“ v souboru „Web.config“, který po každé aktualizaci NET Genia na novou verzi nastavuje program „FinishUpdate.exe“.
- **Framework.txt** – soubor vytvářený automaticky se zjištěnou verzí .NET Frameworku, který je nastaven ve vlastnostech aplikačního poolu přidruženého k webové aplikaci NET Genia
- **Framework4.txt** – existence tohoto souboru zajistí, že bude aktualizací program „Update.exe“ stahovat z internetu nové verze NET Genia určené pro .NET Framework verze 4
- **GoogleTranslationKey.txt** – API klíč ke službě Google Translation API, využívaný v richtextboxech pro překládání textů
- **GoogleTranslationLanguages.txt** – seznam dvoupísmenných zkratk jazyků oddělených středníkem, které jsou nabízeny v richtextboxech pro překládání textů
- **Holidays.txt** – podmínka, podle které budou filtrovány načítané databázové záznamy státních svátků do datových pickerů a do funkcí „ADDWORKDAYS“, „MEASUREWORKDAYS“, „MEASUREWORKHOURS“ nebo „jsMeasureWorkDays“ – například „ng_country = 'USERDETAILS(ng_countryforholidays)'“
- **License.txt** – licenční klíč NET Genia
- **LoginByIdentity.txt** – existence tohoto souboru zajistí, že bude součástí přihlašovacího okna do NET Genia zaškrtnuté tlačítko určené pro automatické přihlášení uživatele v „Active Directory“ pomocí integrovaného ověřování Windows bez nutnosti ručně zadávat přihlašovací jméno a heslo. Spolu s vytvořením tohoto souboru je nutné nakonfigurovat v IIS webovou stránku „LoginByIdentity.aspx“, aby podporovala pouze integrované ověřování Windows, viz. příručka pro instalaci NET Genia.
- **LoginByMicrosoft.txt** – existence tohoto souboru zajistí, že bude součástí přihlašovacího okna do NET Genia zaškrtnuté tlačítko určené pro automatické přihlášení uživatele v „Azure Active Directory“ pomocí integrovaného ověřování Microsoft bez nutnosti ručně zadávat přihlašovací jméno a heslo. Spolu s vytvořením tohoto souboru je nutné nakonfigurovat NET Genium na portálu „https://portal.azure.com“:
 - Registrace aplikace „NET Genium“ na webové stránce „https://portal.azure.com/#blade/Microsoft_AAD_IAM/ActiveDirectoryMenuBlade/RegisteredApps“ (portal.azure.com / Azure Active Directory / Registrace aplikací)
 - **NET Genium** jako název aplikace
 - Zjištění **ID aplikace (klienta)**, který je automaticky vytvořen při registraci aplikace
 - Nastavení URL NET Genia na hodnotu https://{url_netgenia}/LoginByMicrosoft.aspx (portal.azure.com / Azure Active Directory / Registrace aplikací / NET Genium / Ověřování/Identifikátory URI pro přesměrování)
 - Vytvoření **tajného kódu klienta** (portal.azure.com / AzureActiveDirectory / Registrace aplikací / NET Genium / Certifikáty a tajné kódy / Nový tajný kód klienta)
 - Nastavení oprávnění API (portal.azure.com / AzureActiveDirectory / Registrace aplikací / NET Genium / Oprávnění rozhraní API / Přidat oprávnění / MicrosoftGraph)
 - Název rozhraní API „Microsoft Graph – User.Read“, typ „Delegováno“

« Aktualizovat | Máte zpětnou vazbu?

Přehled

- Rychlý start
- Asistent integrace

Správa

- Branding
- Ověřování
- Certifikáty a tajné kódy
- Konfigurace tokenů
- Oprávnění rozhraní API
- Zpřístupnění rozhraní API
- Role aplikací | Preview

Nakonfigurovaná oprávnění

Když aplikacím udělí uživatelé nebo správci v rámci procesu souhlasu oprávnění, mají autorizaci volat rozhraní API. Seznam nakonfigurovaných oprávnění by všechna oprávnění, která aplikace potřebuje. [Další informace o oprávněních a souhlasu](#)

+ Přidat oprávnění Udělit souhlas správce pro NetGenium s.r.o.

Název rozhraní API / oprávnění	Typ	Popis	Vyžaduje se souhlas...	Stav
▼ Microsoft Graph (4)				
Calendars.ReadWrite	Aplikace	Read and write calendars in all mailboxes	Ano	Uděleno pro
Group.ReadWrite.All	Aplikace	Read and write all groups	Ano	Uděleno pro
User.Read	Delegováno	Sign in and read user profile	-	Uděleno pro
User.Read.All	Aplikace	Read all users' full profiles	Ano	Uděleno pro

o Vytvoření konfiguračního souboru „**MicrosoftOAuth.json**“

▪ Obsah souboru nastavit na:

```

{
  "web": {
    "client_id": "ID aplikace (klienta)",
    "auth_uri": "https://login.microsoftonline.com/organizations/oauth2/v2.0/authorize",
    "token_uri": "https://login.microsoftonline.com/organizations/oauth2/v2.0/token",
    "client_secret": "tajný kód klienta"
  }
}
    
```

- **MaxRequestCount.txt** – maximální počet requestů jednoho uživatele z jedné IP adresy za vteřinu. Navýšení výchozího limitu 100 requestů za vteřinu je aktuální především u anonymních webových formulářů, kde všichni uživatelé vystupují v kontextu jednoho nepřihlášeného uživatele.
- **MaxRequestLength.txt** – maximální velikost requestu uživatele uváděná v MB, která se používá pro navýšení výchozího limitu 50 MB, typicky při potřebě pravidelného uploadování velkých souborů do databázových ovládacích prvků „File“ a „Image“ v editačním formuláři. Velikost requestu má vliv na hodnoty atributů „maxRequestLength“ a „maxAllowedContentLength“ v souboru „Web.config“, které po každé aktualizaci NET Genia na novou verzi nastavuje program „FinishUpdate.exe“.
- **MetaTags.txt** – seznam HTML tagů, které se vkládají do každého editačního formuláře do hlavičky webové stránky mezi tagy „<head>“ a „</head>“
- **NoFrames.txt** – existence tohoto souboru zajistí, že bude používána bezrámová verze NET Genia – záhlaví a navigační oblast budou součástí HTML kódu portletů, nahlížecích stránek nebo editačních formulářů
- **NoScript.txt** – textová zpráva, která se automaticky vkládá dovnitř tagu „<noscript>“ do každého editačního formuláře, a určuje zástupný text pro uživatele, kteří nemají povolený javascript ve svém webovém prohlížeči. Například:
 „Vážení uživatelé,
 Váš webový prohlížeč bohužel nepodporuje javascript, který je nutný pro správné zobrazení těchto stránek.
 Prosíme zapněte si jej.
 Děkujeme“
- **OCRServer.txt** – URL vzdáleného NET Genia, které poskytuje služby OCR ve spojení s proměnnou „#ocrserver#“ a funkcemi „OCR2TEXT“ nebo „OCR2FILE“.
- **ResponseHeaders.txt** – seznam hlaviček oddělených enterem, které se automaticky vkládají do každé response. Na každé řádce souboru je umístěna definice jedné hlavičky v následujícím formátu:
 „key: value“
- **SecureCookies.txt** – existence tohoto souboru zajistí, že bude mít každá cookie nastaven parametr „Secure“. Tímto konfiguračním souborem je nutné řídit chování parametru „Secure“ pouze u webových serverů, které mají nastavenou

terminaci SSL na firewallu (například F5). Webové servery, které mají SSL certifikát nastavený přímo v IIS, tento konfigurační soubor nepotřebují, a parametr „Secure“ je každé cookie nastavován automaticky.

- **SessionTimeout.txt** – délka v minutách, po jejímž uplynutí dojde k ukončení session aktuálně přihlášeného uživatele z důvodu jeho neaktivity
- **SingleLogin.txt** – existence tohoto souboru zajistí, že bude mít uživatel aktivní vždy pouze poslední platné přihlášení do aplikace. U předchozích session dojde k automatickému odhlášení.
- **SimpleClick.txt** – existence tohoto souboru zajistí, že bude pro otevírání záznamů z nahlížečích tabulek používáno jednoduché kliknutí. Výchozí nastavení pro otevírání záznamů z nahlížečích tabulek vyžaduje dvojité kliknutí.
- **SSL.txt** – existence tohoto souboru zajistí, že bude každý request přes nezabezpečený kanál „http“ automaticky přesměrován na „https“. Soubor může volitelně obsahovat URL stránky, na kterou bude uživatel přesměrován v případě pokusu o zobrazení NET Genia přes kanál „http“.
- **SysLog.txt** – IP adresy „SysLog“ serverů oddělených středníkem, na které budou automaticky odesílána všechna logová hlášení, viz. kapitola „Logování“
- **Tester.txt** – existence tohoto souboru zajistí, že bude aktualizací program „Update.exe“ stahovat z internetu nové testovací verze NET Genia, které jsou určeny pro partnery nebo vývojáře společnosti NetGenium za účelem ladění nových funkcí. Testovací verze mohou obsahovat chyby, nebo mohou být nestabilní!
- **UnderConstruction.txt** – existence tohoto souboru zajistí, že nebude možné používání webové aplikace, a každý request bude automaticky přesměrován na webovou stránku „UnderConstruction.aspx“ s hlášením o nedostupnosti aplikace. Soubor může volitelně obsahovat textovou zprávu, která se zobrazí namísto výchozího hlášení o nedostupnosti.
- **UnderConstruction2.txt** – existence tohoto souboru zajistí, že nebude možné přihlášení do aplikace nikomu jinému než uživateli „Administrator“ s nejvyšším oprávněním, a každý request všech ostatních uživatelů bude automaticky přesměrován na webovou stránku „UnderConstruction.aspx“ s hlášením o nedostupnosti aplikace. Soubor může volitelně obsahovat textovou zprávu, která se zobrazí namísto výchozího hlášení o nedostupnosti.
- **Version.txt** – číslo aktuální verze NET Genia
- **XFrameOptions.txt** – hodnota hlavičky „X-Frame-Options“, která se vkládá do každé response na hlavní stránce, nahlížečích stránkách, nebo v editačních formulářích. Hlavička „X-Frame-Options“ umožňuje sdělit webovému prohlížeči, zda může být webová stránka zobrazena v rámu. Ve výchozím nastavení není možné výše uvedené stránky zobrazovat v rámu, nastavením hodnoty na „*“ je možné zobrazení v rámu povolit i jiným doménám.

20.3 Souborové přílohy

- Souborové přílohy jsou uloženy na disku v adresáři „Files“. Každá souborová příloha má informaci o názvu souboru uloženou v databázi v tabulce „sfiles“, a na disku je uložena pod změněným názvem souboru. Název souboru na disku bez přípony je definován jako ID záznamu souborové přílohy v tabulce „sfiles“, přípona pak zůstává zachována. Příklad první souborové přílohy uložené v NET Geniu může být například „1.pdf“.
- Souborové přílohy vznikají uploadem souboru pomocí ovládacích prvků „File“ nebo „Image“ v editačním formuláři, vytvořením souborové přílohy serverovou funkcí ve skriptu apod.
- Adresář se souborovými přílohami je možné přesměrovat do jiného umístění na disku pomocí konfiguračního souboru „Files“ bez přípony, který je uložen napřímo ve stejnojmenném adresáři podobně jako bývají ostatní souborové přílohy. Obsah souboru „Files“ pak definuje cestu na disku k alternativnímu adresáři, do kterého mají být souborové přílohy ukládány.

21 Reporty

- Ikona pro vyvolání dialogu s nabídkou dostupných reportů je umístěna v toolbaru v navigační oblasti v administrátorském režimu. Toolbar se zobrazí kliknutím na ikonu ozubeného kolečka, nebo na nápis NET Genium umístěný napravo vedle této ikony.
- Detailní popis reportů je uveden v samostatné příručce „Reporty“.

22 Logování

22.1 Logování incidentů

- Logování incidentů zajišťují dvě základní operace:
 - Ukládání logových hlášení do souborů na disk do adresáře „NETGenium\Logs“
 - Současné odesílání stejných logových hlášení na „SysLog“ server

22.1.1 Logové soubory

- Logová hlášení se ukládají v adresáři „NETGenium\Logs“ do následujících souborů:
 - Auth-year-month.log – Seznam úspěšných a neúspěšných přihlášení a odhlášení
 - Download-year-month.log – Seznam stažených souborů
 - Errors.log – Seznam vzniklých chyb během práce uživatelů
 - Export-year-month.log – Seznam uživatelsky exportovaných dat
 - FinishUpdate.log – Seznam vzniklých chyb během aktualizace aplikace na novou verzi
 - Fulltext-year-month.log – Seznam vzniklých chyb při fulltextovém prohledávání souborových příloh
 - ResendEmlFiles-year-month.log – Seznam příkazů komunikace se SMTP serverem při pokusu o opětovné odeslání e-mailových zpráv z fronty na disku, resp. z adresáře „NETGenium\Logs“
 - RunScript-year-month.log – Seznam vykonaných skriptů prostřednictvím aplikace „RunScript.exe“
 - Scripts60sec-year-month.log – Seznam skriptů, které trvaly déle než 60 vteřin
 - SMTP-year-month.log – Seznam úspěšně i neúspěšně odeslaných e-mailů
 - System.log – Seznam informativních hlášek o spuštění aplikace (často souvisí se spuštěním nebo restartem aplikačního poolu)
 - UnsafeTags.log – Seznam původně zadaného a výsledného HTML kódu, který byl vyčištěn během zadávání formátovaných textů u ovládacích prvků „RichTextBox“
 - Version.log – Seznam změn databázové struktury souvisejících s novou verzí aplikace
 - Warnings.log – Seznam upozornění
- Adresář s logovými soubory je možné přesměrovat do jiného umístění na disku pomocí konfiguračního souboru „Logs“ bez přípony, který je uložen napřímo ve stejnojmenném adresáři podobně jako bývají ostatní logové soubory. Obsah souboru „Logs“ pak definuje cestu na disku k alternativnímu adresáři, do kterého mají být logové soubory ukládány.

22.1.2 SysLog

- Odesílání logových hlášení je možné nastavit na více „SysLog“ serverů zároveň. Nastavení je provedeno pomocí konfiguračního souboru „SysLog.txt“ umístěného v adresáři „NETGenium\Config“.
- Pokud je konfigurační soubor „SysLog.txt“ vytvořen, a je v něm uvedena jedna IP adresa „SysLog“ serveru, nebo více IP adres „SysLog“ serverů oddělených středníkem, budou automaticky všechna logová hlášení odesílána na uvedené „SysLog“ servery, a zároveň budou ukládána do adresáře „NETGenium\Logs“.
- Při absenci konfiguračního souboru „SysLog.txt“ je logování prováděno pouze do souborů v adresáři „NETGenium\Logs“.

- Zprávy odesílané na „SysLog“ server mají následující formát, který používá mezeru jako oddělovač jednotlivých hodnot: „datum čas doménový_název_serveru aplikace kategorie typ_události přihlašovací_jméno_uživatele ip_adresa_uživatele zpráva“.
 - Kategorie – Auth, Download, Errors, atp.
 - Typ události – VERBOSE, NOTICE, WARNING, ERROR, FATAL

22.2 Logování rychlosti webové aplikace

- NET Genium loguje do databáze následující události:
 - Rychlost načtení editačních formulářů – do databázové tabulky „stables_stats“
 - Rychlost načtení nahlížečích stránek – do databázové tabulky „sviewpages_stats“
 - Rychlost databázových dotazů – do databázové tabulky „squerybuilder_stats“
 - Rychlost externích funkcí – do databázové tabulky „snggef_stats“
- Zapnutí nebo vypnutí logování se provádí v samostatném dialogu pod odkazem „Upravit NET Genium“ na záložce „Ostatní“:
 - Vypnuté – logování je vypnuté
 - Do databáze – logování probíhá pouze do databáze
 - Na disk – logování probíhá pouze na disk do adresáře „Logs“
 - Do databáze a na disk – logování probíhá do databáze a na disk do adresáře „Logs“
- Zobrazení agregovaných statistik z logů vytvořených v databázi je možné pomocí reportů, viz příručka „Reporty“.

23 Instalace

- Obvyklá instalace NET Genia se skládá z následujících kroků:
 - Instalace databázového serveru Firebird nebo MSSQL
 - Detailní popis instalace databázového serveru Firebird je uveden v samostatné příručce „Instalace Firebird 3.0“.
 - Detailní popis instalace databázového serveru MSSQL je uveden v samostatné příručce „Instalace MSSQL“.
 - Instalace NET Genia – detailní popis instalace NET Genia je uveden v samostatné příručce „Instalace NET Genium“.
 - Instalace SMTP serveru – detailní popis instalace SMTP serveru je uveden v samostatné příručce „Instalace SMTP“.
- Nasazení NET Genia pro kritické aplikace by mělo rozlišovat vývojové a provozní prostředí, provozované ideálně na dvou rozdílných serverech.
- Detailní popis správy vývojového prostředí a vydávání nových release je uveden v samostatné příručce „Vývojové prostředí“.

24 Utility

- NET Genium obsahuje řadu utilit umístěných na disku v adresářích „Backup“, „bin“ a „Config“:
 - Backup
 - BackupServerService.exe
 - BackupServer.exe
 - FileBackup.exe
 - SqlBackup.exe
 - bin
 - ApplicationManager.exe
 - FileUpload.exe
 - FinishUpdate.exe
 - LogService.exe
 - OnlineUsers.exe
 - PrintPdf.exe
 - ResendEmlFiles.exe
 - Restart.exe
 - RunningQueries.exe
 - RunScript.exe
 - Setup.exe
 - Config\Tools
 - Deactivate.exe
 - Activate.exe
 - GrantLogin.sql
 - GrantLogin.bat.txt
 - MemoryDumps.txt
 - SSL.reg
 - SSL-ie6.reg
 - TracingRequests.txt
 - TuningQueries.sql
 - TuningQueries.txt
 - WALTU.exe
- Detailní popis všech utilit je uveden v samostatné příručce „Utility“.

25 Zálohování

- Detailní popis zálohování NET Genia je uveden v samostatné příručce „Zálohování“.