



## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

<b>Název:</b>	Koncept webové aplikace pro spolehlivostní modely
<b>Student:</b>	Jaroslav Jasenovský
<b>Vedoucí:</b>	Ing. Martin Daňhel, Ph.D.
<b>Studijní program:</b>	Informatika
<b>Studijní obor:</b>	Informační systémy a management
<b>Katedra:</b>	Katedra softwarového inženýrství
<b>Platnost zadání:</b>	Do konce letního semestru 2020/21

### Pokyny pro vypracování

Nastudujte teorii spolehlivosti (rozsah vymezí vedoucí práce) zaměřte se zejména na problematiku popisu systémů pomocí HDM.

Analyzujte možnosti, jak uživatelsky přívětivě vytvářet HDM pomocí webového rozhraní, předpokládejte možnost reálného nasazení.

Dle analýzy vytvořte koncept webové aplikace pro popis systémů pomocí HDM modelů z nastudovaných zdrojů, dle těchto kroků:

1. Analyzujte potřebné uživatelské role. Uživatelé by měli mít možnost spolupracovat na projektu, kde projekt může znamenat i více modelů, projektem se dále rozumí popis konkrétního systému spolehlivostním modelem, či slovním zadání apod.
2. Dále analyzujte vhodné prostředky pro vývoj aplikace.
3. Aplikace bude umožňovat autorizaci uživatelů, správu (vytváření/mazání/editaci) projektů. Předpokládá se návrh základního GUI pro snadnou manipulaci s modely, projekty a uživateli.

Výstupem bakalářské práce bude studie proveditelnosti reálného nasazení.

### Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.  
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.  
děkan

V Praze dne 9. prosince 2019





**FAKULTA  
INFORMAČNÍCH  
TECHNOLGIÍ  
ČVUT V PRAZE**

Bakalářská práce

# **Koncept webové aplikace pro spolehlivostní modely**

*Jaroslav Jasenovský*

Katedra Číslicového návrhu

Vedoucí práce: Ing. Martin Daňhel, Ph.D.

30. července 2020



---

## Poděkování

Chtěl bych poděkovat Ing. Martinu Daňhelovi, Ph.D. za vedení mé bakalářské práce, cenné rady a odborný dohled. Děkuji také Ing. Petře Pavlíčkové, Ph.D. za pomoc při strukturování práce a tvorbě prezentace. Dále děkuji Bc. Janě Menšíkové za podporu a gramatickou kontrolu práce.



---

## Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principu při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisu, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 citovaného zákona.

V Praze dne 30. července 2020

.....

České vysoké učení technické v Praze  
Fakulta informačních technologií

© 2020 Jaroslav Jasenovský. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Jasenovský, Jaroslav. *Koncept webové aplikace pro spolehlivostní modely*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2020.



---

# Abstrakt

Cílem bakalářské práce bylo navrhnout webovou aplikaci, která umožňuje vývoj hierarchických spolehlivostních modelů. Navzdory tomu, že tyto modely představují užitečný nástroj pro výpočet spolehlivosti systémů, nejsou zatím široce podporovány. Dále bylo třeba vypracovat pro tuto aplikaci studii proveditelnosti, která má zhodnotit její přínosy pro technicky zaměřené společnosti a školy. Toho bylo docíleno vyhodnocením aktuálního stavu a porovnáním procesů s navrženým řešením aplikace. Správný návrh databáze a uživatelských rolí byl hlavním předpokladem pro snadnou skupinovou práci, což vedlo ke zvýšení efektivity práce. Dobře navržené a přívětivé uživatelské rozhraní je dalším prvkem, který uživateli značně usnadní práci v modeláři, a urychlí tak samotný proces vývoje. Díky tomu, že aplikace je navržena nativně pro tvorbu hierarchických spolehlivostních modelů, podporuje na rozdíl od jiných značnou automatizaci mnoha procesů při vývoji modelu. Kombinace všech těchto aspektů umožní značné usnadnění při vývoji spolehlivostních modelů, a tím i samotný návrh systémů.

**Klíčová slova** Hierarchické spolehlivostní modely, návrh webové aplikace, studie proveditelnosti, online modelář, uživatelské skupiny, uživatelské role, databáze

# Abstract

The goal of this work was to design a web application which would enable the development of Hierarchical Dependability Models (HDM). Although these models represent an efficient tool for system reliability prediction, they are still not supported enough. Our next task was to make a feasibility study of this application to appraise its benefits to technology-oriented companies and schools. It was carried out by evaluating the current state and comparing processes with our web application design. A good database and user roles design has been a crucial precondition for smooth teamwork which has led to higher efficiency. Well-designed and user-friendly GUI is another important aspect which makes working with the modeller easier and thus speeds up the process of development itself. Since the application is designed originally to develop HDM, unlike the others, it supports the automatization of many development processes. Combination of all these aspects provides a remarkable facilitation of HDM development, hence it fastens the system design.

**Keywords** Hierarchical Dependability Models, web application design, feasibility study, on-line modeller, users' group, user role, database

---

# Obsah

<b>Úvod</b>	<b>1</b>
<b>1 Cíl práce</b>	<b>3</b>
<b>2 Spolehlivost Systému</b>	<b>5</b>
2.1 Hierarchické spolehlivostní modely . . . . .	7
<b>3 Analýza a Návrh</b>	<b>11</b>
3.1 Specifikace požadavků . . . . .	11
3.2 Analýza uživatelských rolí . . . . .	12
3.3 Analýza uživatelského rozhraní . . . . .	14
3.3.1 Návrh uživatelského rozhraní . . . . .	14
3.4 Analýza použitých technologií . . . . .	19
3.5 Relační návrh aplikace . . . . .	20
<b>4 Studie Proveditelnosti</b>	<b>23</b>
4.1 Souhrnný obsah struktury studie proveditelnosti . . . . .	23
4.2 Úvodní informace . . . . .	24
4.3 Stručný popis podstaty projektu a jeho etap . . . . .	24
4.4 Analýza trhu . . . . .	26
4.5 Technické a technologické řešení projektu . . . . .	29
4.6 Hodnocení efektivity a udržitelnosti projektu . . . . .	31
4.6.1 Tvorba modelu . . . . .	32
4.6.2 Spolupráce na modelu . . . . .	36
4.6.3 Správa skupiny . . . . .	39
4.6.4 Náhled modelu: . . . . .	41
4.6.5 SWOT analýza . . . . .	43
4.7 Řízení rizik . . . . .	43
4.7.1 Analýza rizik . . . . .	44

4.7.2	Ošetření rizik . . . . .	45
4.7.2.1	Nepřijetí aplikace - R1 . . . . .	45
4.7.2.2	Zkušenosti - R3 . . . . .	46
4.7.2.3	Nemoc či zranění - R4 . . . . .	46
4.7.2.4	Syndrom vyhoření - R2 . . . . .	46
4.8	Závěrečné hodnocení projektu . . . . .	46
	<b>Závěr</b>	<b>49</b>
	<b>Literatura</b>	<b>51</b>
	<b>A Seznam použitých zkratk</b>	<b>53</b>
	A.1 Slovník pojmů . . . . .	53
	<b>B Obsah příloženého CD</b>	<b>55</b>

---

## Seznam obrázků

2.1	Ukázka kombinovaného modelu systému. Prvky B0-B3 jsou vůči sobě v paralelním zapojení a tvoří celek, který je s prvky B3-B5 v sériovém zapojení.[2]	5
2.2	Stromu poruch, kde je prvek f tvořen prvky f1-f3, které jsou spojeny hradlem OR, a tudíž selhání jakéhokoliv prvku f1-f3 vede k selhání f.[2]	6
2.3	Markovův model se třemi stavy. První stav OK značí, že systém je v pořádku. Stav 1F je již poruchový a značí poruchu jednoho nezávislého modulu, ale systém se z něho může vzpamatovat. Poslední stav F je poruchový a koncový. Proměnné $a$ a $b$ slouží k výpočtu pravděpodobnosti přechodu mezi stavy.[2]	7
2.4	Graf znázorňující hierarchii v běžných elektronických systémech.[2]	8
2.5	Ukázka způsobu převedení blokového modelu do HDM. Prvky v HDM, které jsou znázorněny červenou barvou (P a S), značí paralelní či sériové zapojení prvků, jež jsou vůči nim hierarchicky níže.[2]	9
2.6	Každý prvek v HDM z minulého obrázku může mít predikovanou spolehlivost jiným způsobem.[2]	10
3.1	Návrh úvodní obrazovky pro přihlášení a registraci uživatele.	15
3.2	Návrh obrazovky uživatelského účtu, kde se nachází hlavní rozcestník pro další činnosti.	16
3.3	Návrh obrazovky zobrazující určitý projekt uživatele a obsahující modely, které do projektu náleží.	17
3.4	Návrh obrazovky zobrazující rozpracovaný model. Na obrazovce se vždy zobrazuje pouze blok daného modelu, aby byla co nejsnazší a přehledná.	18
3.5	Návrh obrazovky pro skupinu, kde se zobrazují všechny projekty skupiny a také je zde nástěnka pro poznámky.	19
3.6	Vrstvy navrhovaného systému s použitými technologiemi.	20

3.7	Relační model webové aplikace znázorňující vztahy jednotlivých objektů v modelu. Slouží pro vytvoření databáze a tříd, jež mají s danými objekty pracovat. Model byl vytvořen v aplikaci Enterprise Architect . . . . .	21
3.8	Grafické znázornění kritických částí v daném bloku modelu. Lze krokovat až k původu dané chyby. . . . .	22
4.1	Životní cyklus vývoje systému, kde je nejdříve potřeba analyzovat požadavky. Následuje návrh aplikace a její vývoj, který je dále testován a konzultován. Pokud se objeví nějaké nové požadavky či problémy, tak se opět vrací do návrhu, jinak je daná část nasazena a jsou řešeny další požadavky. . . . .	26
4.2	Ukázka vygenerovaného grafu, pro složitější problém.[8] . . . . .	28
4.3	Ukázka bloku v HDM, který je je zapojen sériově a skládá se z jednoduchých součástí (cívka, dioda). . . . .	30
4.4	BPMN model tvorby HDM v aktuálním řešení, kdy neexistuje aplikace podporující HDM. Model byl vyvíjen v aplikaci Bizagi modeler. . . . .	33
4.5	BPMN model tvorby HDM v navrhovaném řešení určeného pro HDM. Model byl vyvíjen v aplikaci Bizagi modeler. . . . .	35
4.6	BORM model procesu sdílení modelu. V aktuálním stavu je třeba používat aplikace třetích stran jako například verzovací nebo zasílací služby. Model byl vyvíjen v aplikaci Openponk. . . . .	37
4.7	BORM model procesu sdílení modelu. V navrhovaném řešení lze celý proces učinit v rámci navrhované webové aplikace. Model byl vyvíjen v aplikaci Openponk. . . . .	38
4.8	BPMN model procesu správy skupiny. Obrázek znázorňuje navrhované řešení, kde je funkce podporována přímo v rámci webové aplikace. Model byl vytvořen v aplikaci Bizagi modeler. . . . .	40
4.9	BPMN model procesu zaslání náhledu na model. Navrhované řešení umožňuje sdílení pomocí odkazu, díky kterému získá daný host přístup read-only. Model byl vytvořen v aplikaci Bizagi modeler. . . . .	42

---

## Seznam tabulek

4.1	Popisná tabulka procesu tvorby modelu. Zde je popsáno aktuální řešení, kde není nativní aplikace pro HDM. . . . .	32
4.2	Popisná tabulka procesu tvorby modelu. Zde je popsáno navrhované řešení webové aplikace. . . . .	34
4.3	Popisná tabulka procesu spolupráce při tvorbě modelu. Zde je popsáno navrhované řešení webové aplikace. . . . .	36
4.4	Popisná tabulka procesu tvorby modelu. Zde je popsáno navrhované řešení webové aplikace. . . . .	38
4.5	Popisná tabulka procesu správy skupiny. Zde je popsáno navrhované řešení webové aplikace. . . . .	39
4.6	Popisná tabulka procesu náhledu modelu. Zde je popsáno navrhované řešení webové aplikace, kde je možné zaslat náhled i uživatelům, jenž nejsou registrovaní ve webové aplikaci. . . . .	41
4.7	Tabulka SWOT analýzy projektu, kde jsou zhodnoceny vnitřní i vnější silné a slabé stránky projektu. . . . .	43
4.8	Matice pravděpodobnosti a dopadu.[11] . . . . .	45





---

# Úvod

V dnešním konzumním světě, a dalo by se říci v době informační, je důležitá především efektivita a produktivita. S tím také souvisí rychlost komunikace, a proto se již téměř nic neobejde bez spolupráce nějakého informačního systému či webové aplikace. Tato práce se bude zabývat návrhem webové aplikace určené pro tvorbu hierarchických spolehlivostních modelů. Tato aplikace by měla usnadnit komunikaci v pracovních týmech i samotný vývoj elektronických systémů. Součástí mé práce je i studie spolehlivosti, která má hodnotit především stránku konzumního světa, a hlavně jakou může tato aplikace přinést efektivitu při práci.

První kapitola je věnována vysvětlení pojmu spolehlivosti systému. Dále se zabývá různými způsoby predikce této spolehlivosti a k tomu využitých modelů, přičemž se snaží čtenáři přiblížit hierarchické spolehlivostní modely (Hierarchical Dependability Models, dále HDM). Tím získá představu o jejich výhodě při výpočtu pravděpodobnosti, že se modelovaný systém dostane do kritického stavu. Tato hodnota je klíčová ke specifikaci Safety Integrity Level (SIL), který slouží k dalším specifikacím systémů. To vše má ukázat smysl podpory těchto modelů, jež umožňují snadno pracovat jak s návrhy těch nejjednodušších, tak nejsložitějších systémů.

Druhá kapitola obsahuje analýzu a návrh webové aplikace. Její důležitou součástí je návrh uživatelských rolí a dalších vzájemných vztahů uživatelů tak, aby co nejvíce usnadňoval týmovou práci na modelech i rozsáhlých projektech, což je klíčové pro zvýšení efektivity. Také zde najdete návrh a ukázkou uživatelského rozhraní, jež má být jednoduché a navržené na míru cílové skupině. Rovněž je zde předvedeno grafické znázornění zvýraznění kritických částí v modelu a jejich následný vliv, což je součástí validace a propagace chyb v modelu a jeho simulace v určitých podmínkách. Dále se zde hodnotí možné použité technologie pro vývoj této aplikace. Nakonec si představíme relační návrh aplikace, který má ukázat vztahy mezi jednotlivými objekty, a slouží pro vývoj databáze a tříd.

Ve třetí kapitole je vypracována studie proveditelnosti, jež má prokázat smysl celého projektu. V této studii je provedena analýza trhu, z níž získáme představu o aktuálním stavu, s kterým budeme srovnávat navrhované řešení webové aplikace. V potaz jsou brána i rizika, u kterých se snažíme minimalizovat vliv na daný projekt.

Rád bych podotkl, že cílem práce není implementace kompletní webové aplikace a že hlavním výstupem je její analýza a návrh, společně se studií proveditelnosti reálného nasazení. Tato studie analyzuje podnikatelský záměr a ověří užitečnost daného projektu. Dále budou díky analýze a návrhu postaveny základy pro budoucí tvorbu této webové aplikace.

Hlavním podnětem pro volbu tohoto tématu je to, že vedoucímu této práce by navrhovaná webová aplikace usnadnila budoucí práci na rozsáhlých systémech, které jsou kritické z pohledu spolehlivosti a bezpečnosti. Finální verze této aplikace by jistě našla uplatnění v mnoha institucích, neboť ulehčuje práci jednotlivce, kterému usnadní přístup a správu projektů, menším skupinám pracujících na jednom modelu či menších projektech, ale i velkým týmům, které nemusí pracovat na téže věci. Těm se může zejména hodit znalost o modelech dalších pracovních skupin a především možnosti mít vůdčí osobu, která má možnost veškerou tuto práci kontrolovat, ať v podobě projektového manažera, či vyučujícího ve školách.

Další motivací jsou určitě zkušenosti, neboť při tvorbě mohu využít všechny své znalosti, které jsem mohl zatím při studiu získat, a dále je rozšířit díky této práci. Také získám ekonomicko-manažerský pohled na projekt, což skvěle zapadá do dnešního smýšlení, protože málokdo investuje do práce, která nebude produkovat žádný budoucí ekonomický zisk.

---

## Cíl práce

Cílem bakalářské práce je vytvořit návrh webové aplikace pro tvorbu HDM a vypracovat pro tuto aplikaci studii proveditelnosti.

Pro studii je nezbytné analyzovat stávající metody vývoje HDM, aby bylo navrhované řešení s čím porovnat s čím porovnat. Díky tomu je možné přehled o vnitřních procesech a jejich zefektivnění v navrhované aplikaci, což vede k následnému zhodnocení ekonomických přínosů především v časové úspoře a usnadnění týmové práce. Po této části bude vycházet již technologický návrh aplikace a důležitých vnitřních struktur. Finální částí bude návrh jednoduchého uživatelského rozhraní, které se bude snažit být co nejvíce uživatelsky přívětivé a samozřejmě se bude soustředit na cílovou skupinu této aplikace, kterou bude také třeba lépe specifikovat.

Výsledky práce budou přínosné pro následnou tvorbu dané aplikace, která umožňuje snadnou spolupráci lidí při tvorbě modelů. Také zhodnotí její ekonomické přínosy v technických oborech a podnicích zabývajících se vývojem systémů.

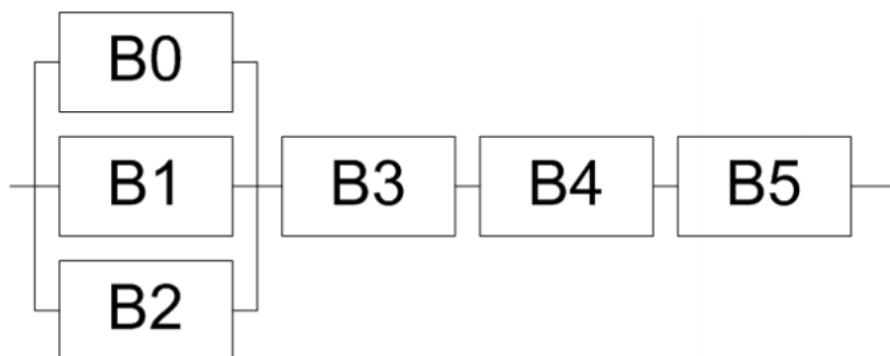


## Spolehlivost Systému

Dle ČSN 010102 je spolehlivost systému charakterizována jako: „Obecná vlastnost objektu spočívající ve schopnosti plnit funkce při zachování hodnot stanovených provozních ukazatelů v daných mezích a v čase podle stanovených technických podmínek.“[1]

Při snaze o predikci spolehlivosti systému je třeba určit hodnoty jeho celkových spolehlivostních ukazatelů ze známých hodnot spolehlivostních ukazatelů prvků a ze způsobu jejich spojení. Pro řešení tohoto problému se využívají spolehlivostní modely.[1]

Dle složitosti systému lze volit z různých typů modelů. Například pro systémy s nezávislými prvky lze použít sériový model, paralelní model či jejich kombinaci. Tyto modely jsou snadné a intuitivní. Všechny tyto modely se společně nazývají blokové modely.[1]

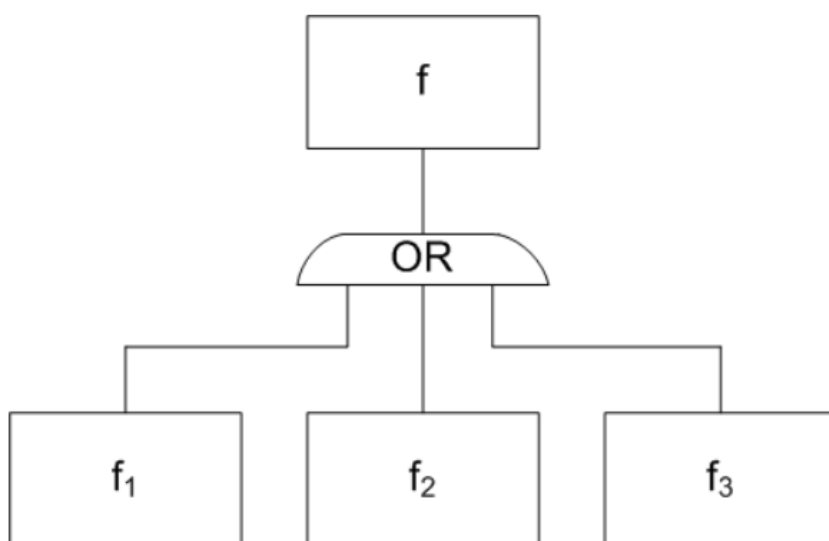


Obrázek 2.1: Ukázka kombinovaného modelu systému. Prvky B0-B3 jsou vůči sobě v paralelním zapojení a tvoří celek, který je s prvky B3-B5 v sériovém zapojení.[2]

## 2. SPOLEHLIVOST SYSTÉMU

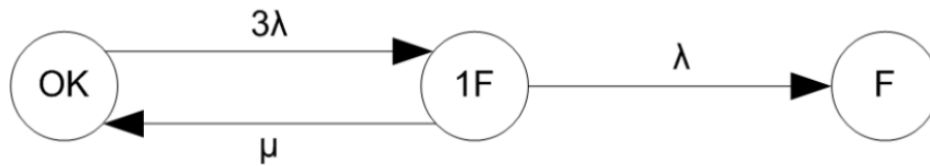
---

V praxi se kvůli složitějšímu členění využívají více modely založené na stavovém diagramu, ovšem jejich problém je rychlý nárůst počtu stavů vůči počtu součástí. Tento problém řeší buď metoda řezu, která bere v potaz pouze souvislé množiny součástí v systému, nebo stromu poruch, kde se systém větví ve stromové struktuře na různé události spojené hradly (nejčastěji AND a OR), které připodobňují sériové a paralelní zapojení.[1]



Obrázek 2.2: Stromu poruch, kde je prvek  $f$  tvořen prvky  $f_1$ - $f_3$ , které jsou spojeny hradlem OR, a tudíž selhání jakéhokoliv prvku  $f_1$ - $f_3$  vede k selhání  $f$ . [2]

Pro ještě složitější případy se využívají například Markovské modely, které jsou založeny na stavovém modelu, kde jsou jednotlivé stavy nezávislé a pracuje se s pravděpodobností přechodu mezi stavy. Výhodou těchto modelů je bohatá aplikační podpora a dobře definované matematické základy. Nevýhodou je nárůst stavů u složitějších modelů a následná složitost výpočtu, jež je časově náročná.[1]



Obrázek 2.3: Markovův model se třemi stavy. První stav OK značí, že systém je v pořádku. Stav 1F je již poruchový a značí poruchu jednoho nezávislého modulu, ale systém se z něho může vzpamatovat. Poslední stav F je poruchový a koncový. Proměnné  $\lambda$  a  $\mu$  slouží k výpočtu pravděpodobnosti přechodu mezi stavy.[2]

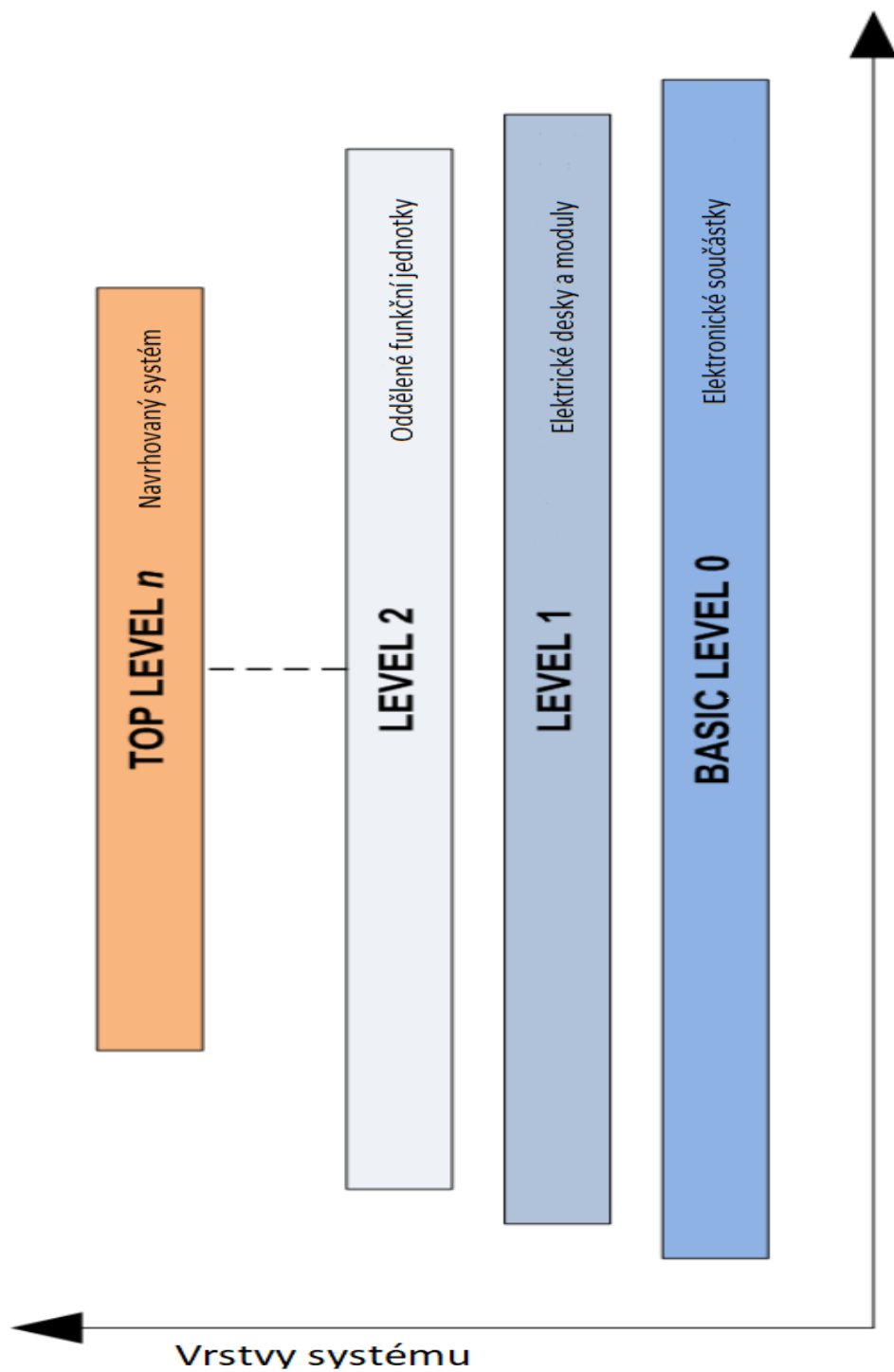
Aby byl zachován vztah mezi jednotlivými částmi, na které se funkční celek rozpadá, je na ně aplikován některý z principů blokových systému. Díky tomu lze sledovat vliv jednotlivých částí směrem vzhůru až ke komplexnímu systému. Prvky v nejnižších vrstvách (listy stromu, použijeme-li grafovou terminologii) jsou součástky, jež mají specifikovány kritické hodnoty a určenou spolehlivost. Mohou to být samostatné elektronické součástky nebo funkční části, které mají spolehlivost predikovanou jiným modelem.[1]

## 2.1 Hierarchické spolehlivostní modely

Pro nejnáročnější systémy se používají víceúrovňové modely, které hlavní systém dekomponují na jednodušší subsystémy, což lze aplikovat opakovaně ve více úrovních dokud nedostaneme minimální část, která má své kritické hodnoty zadány (např. diody, cívky, tranzistory,...). Do této kategorie spadají i HDM, pro něž je v této práci navrhována webová aplikace, aby byla práce s nimi usnadněna. Tyto modely kombinují blokové modely a stromy poruch. Využívají skutečnosti, že většina dnešních modelů je navrhována hierarchicky v jednotlivých úrovních. Z toho plyne, že každý systém může být rozdělen na menší funkční celky. To může být v závislosti na složitosti aplikováno opakovaně, dokud není daná část složena z jednotlivých elektronických součástek.[2]

## 2. SPOLEHLIVOST SYSTÉMU

---

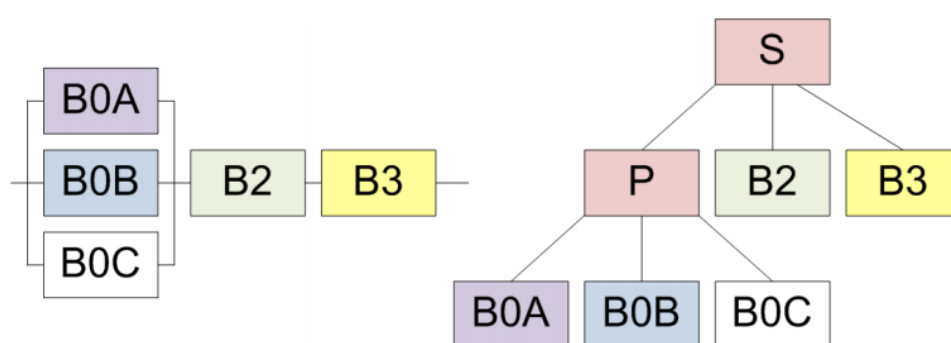


Obrázek 2.4: Graf znázorňující hierarchii v běžných elektronických systémech.[2]



## 2.1. Hierarchické spolehlivostní modely

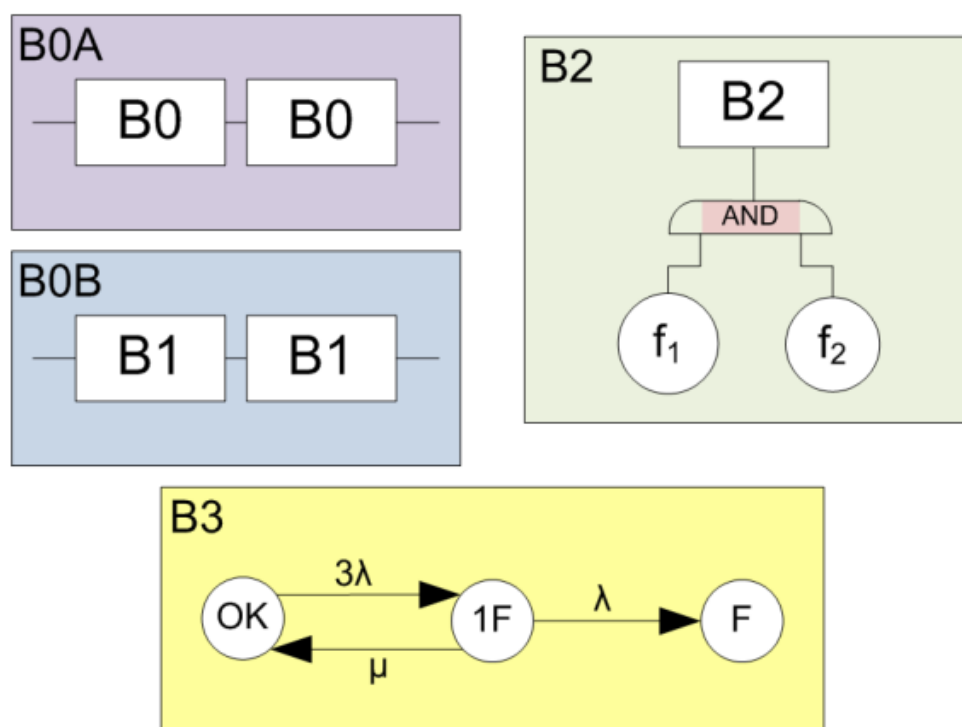
Aby byl zachován vztah mezi jednotlivými částmi, na které se funkční celek rozpadá, je na ně aplikován některý z principů blokových systému. Díky tomu lze sledovat vliv jednotlivých částí směrem vzhůru až ke komplexnímu systému. Prvky v nejnižších vrstvách (listy stromu, použijeme-li grafovou terminologii) jsou součástky, jež mají specifikovány kritické hodnoty a určenou spolehlivost. Mohou to být samostatné elektronické součástky nebo funkční části, které mají spolehlivost predikovanou jiným modelem.[2]



Obrázek 2.5: Ukázka způsobu převedení blokového modelu do HDM. Prvky v HDM, které jsou znázorněny červenou barvou (P a S), značí paralelní či sériové zapojení prvků, jež jsou vůči nim hierarchicky níže.[2]

## 2. SPOLEHLIVOST SYSTÉMU

---



Obrázek 2.6: Každý prvek v HDM z minulého obrázku může mít predikovanou spolehlivost jiným způsobem.[2]

Výhody těchto modelů jsou jednoduchost a rychlost zpracování. Lze ho použít pro většinu systémů a uživatel si může navrhnout vlastní pravidla v bloku. Kritické části lze nalézt velmi rychle. Nevýhodami je chybějící aplikační podpora a nemožnost modelovat stavy systému.[2]

---

## Analýza a Návrh

### 3.1 Specifikace požadavků

Je potřeba vytvořit webovou aplikaci pro tvorbu Hierarchical Dependency Models (HDM), která tak usnadní a hlavně urychlí práci ve vývoji systémů.

Tato aplikace je určena pro potřeby škol, jak k potřebám výzkumných týmů, tak pro potřeby výuky, ale bylo by možné nasazení i do technických firem.

Důležitou vlastností navrhované aplikace bude kromě samotného vytváření modelů také jejich automatická kontrola a propagace kritických hodnot v dané hierarchii modelu. Další potřebou v tomto oboru je i týmová práce. A to jak v projektech, které obsahují jen jeden model, tak i v projektech, které se skládají z většího množství týmů, které pracují na různých systémech. Také je zde studijní potřeba, aby mohl mít vyučující kontrolu nad projekty svých studentů.

Zásadní procesy:

- Vývoj modelu
- Spolupráce na modelu
- Náhled model
- Správa skupin

Z tohoto popisu je vidět, že tyto procesy se dají zařadit do dvou skupin. První je práce ve vlastním projektu s modely, kde bude možnost si do této práce pozvat nějaké spolupracovníky či ukázat nějakému konzultantovi daný model. A druhá se zabývá sjednocením těchto jednotlivců či spolupracovníků do větších týmů, ať už pracovních nebo studentských, kde je nějaká zastřešující role buď vyučujícího, nebo třeba projektového manažera či teamleadera.

Díky tomu, že se tyto procesy spojí do jedné webové aplikace, může probíhat živá spolupráce. Usnadní se značně rychlost tvorby těchto modelů, neboť

aktuálně vývoj probíhá buď formou práce u jednoho počítače, nebo vytvoření modelu a jeho následné zaslání v nějaké přenosné podobě kolegovi. Další usnadnění práce je, že aplikace bude určena přímo pro HDM, a tudíž může podporovat i kontrolu modelu a dělat výpočty za uživatele a například graficky zvýrazňovat kritické části v modelu. Dále podpora vytváření skupin s jednou vedoucí osobou umožní snadné využití pro studijní i korporátní potřeby, a díky tomu usnadní i veškerý vývoj ve větších týmech. A finální výhodou, kterou nelze opomenout, je i možnost práce z domova díky webovému rozhraní, a to je plus v obou cílových skupinách.

## 3.2 Analýza uživatelských rolí

Základem každého multi-uživatelského systému je určit si rozdělení rolí všech uživatelů tak, aby měl každý práva pro svou činnost, ale neměl možnost zasahovat do práce ostatním uživatelům. Což by se dalo udělat jednoduše oddělením jednotlivých uživatelských prostředí. V této aplikaci je však základním kamenem projekt, na kterém ovšem může pracovat více uživatelů najednou, proto je třeba navrhnout samotný systém závislostí vůči jednotlivým objektům.

Bez delších úvah je jasné, že v tomto systému musí existovat role *správce*. Tato role dává uživateli veškerá práva a je určena pro samotnou správu aplikace, a to i v případě, že nějaký jiný uživatel učiní kroky, které jsou pro jeho roli již nezvratné.

Další rolí je *registrovaný uživatel*. Tuto roli získá každý uživatel, který se registruje a vytvoří si tak uživatelský účet. Tato role umožňuje uživateli zcela bez omezení pracovat na svém “písečku” (uživatelském účtu).

Tato hierarchie je základní kostrou a role jsou přiděleny hned při přihlášení daných uživatelů. Zbylé role vznikají až v průběhu práce a jsou vždy vázány na určitý objekt v aplikaci (skupina, project, ...). Toto vede k tomu, že každý uživatel bude mít v jeden okamžik vždy jednu hlavní roli a může mít nadále nula až desítky vázaných rolí.

První z těchto rolí je vázána k projektu, který si může každý uživatel vytvořit. Po jeho vytvoření se tak stává *vlastníkem projektu*. V daném projektu pak může libovolně tvořit modely a má zde plná práva. Ovšem, protože je třeba zajistit možnost pracovat na jednom projektu ve více lidech, má každý vlastník projektu možnost přizvat k němu další registrované uživatele. Tito uživatelé dostanou roli *spolupracovníka*, díky níž mají možnost plně pracovat s modely v daném projektu. Jediným omezením je schvalovací režim při mazání modelu, kdy pokud chce nějaký spolupracovník smazat celý model v projektu, musí tuto akci nejprve schválit vlastník projektu (do té doby bude akce viditelná pouze pro žadatele o smazání).

Další vázanou roli dostane uživatel v momentě, kdy vytvoří pracovní skupinu, a stane se tak *správce skupiny*. Tento objekt je tu pro účely práce

ve školách či firmách, kdy potřebujeme spojit několik uživatelů, kteří ovšem budou pracovat na různých projektech. Vzhledem k tomu, že správce skupiny (může to být teamleader, project manager, učitel, ...) bude mít plnou kontrolu nad projekty uživatelů ve skupině, je třeba tyto projekty odlišit od ostatních, aby nemohl správce skupiny zasahovat do všeho, co daný uživatel dělá. Je tak u každého vytvořeného projektu možnost připojit ho ke skupině, již je daný uživatel členem. Díky tomuto je tak možné velké spolupráce v rámci chráněných území uživatelů. Například učitel vytvoří skupinu, kam pozve studenty z jedné třídy, a může tak kontrolovat jejich práci a přidávat k ní své poznámky. A zároveň si mohou studenti vytvořit pracovní skupinky, kdy jeden založí projekt a další studenty k němu přizve. Stejně to lze interpretovat i v rámci firmy.

Poslední rolí je *Host*, který není registrovaným uživatelem, a můžeme ho pozvat přes link zasláný na emailovou adresu, přes kterou je i daný uživatel dočasně identifikován, aby bylo možné i jeho činnost nadále spravovat. Host má pouze práva náhledu do daného modelu. Tuto roli, lze přidělit i registrovanému uživateli, kde se již nespojuje přes emailovou adresu, ale přímo k jeho účtu, a jde o roli vázanou k modelu.

Správce je určen při spuštění webové aplikace. Každý další se může registrovat jako uživatel, jehož role je spojena vůči dalším objektům v aplikaci. Může si tudíž vytvořit pracovní skupinu, kde má roli správce. Může být pozván do skupiny, pak když vytvoří projekt, kde má práva vlastníka projektu, může projekt přidat k dané skupině, aby v něm měl potřebná práva i správce skupiny. A může si přizvat k projektu spolupracovníka. Příklad hosta je použitelný, pokud by uživatel chtěl ukázat nějaký model další osobě, ale nechce jí dát možnost upravovat a není nutnost registrace dané osoby; například chce-li poslat svou práci ke schválení zákazníkovi.

- Správce
- Registrovaný uživatel
  - Projekt
    - \* Vlastník projektu
    - \* Spolupracovník
  - Skupina
    - \* Vlastník skupiny
    - \* Běžný uživatel
  - Model
    - \* Vlastník projektu
    - \* Spolupracovník
    - \* Host

## 3.3 Analýza uživatelského rozhraní

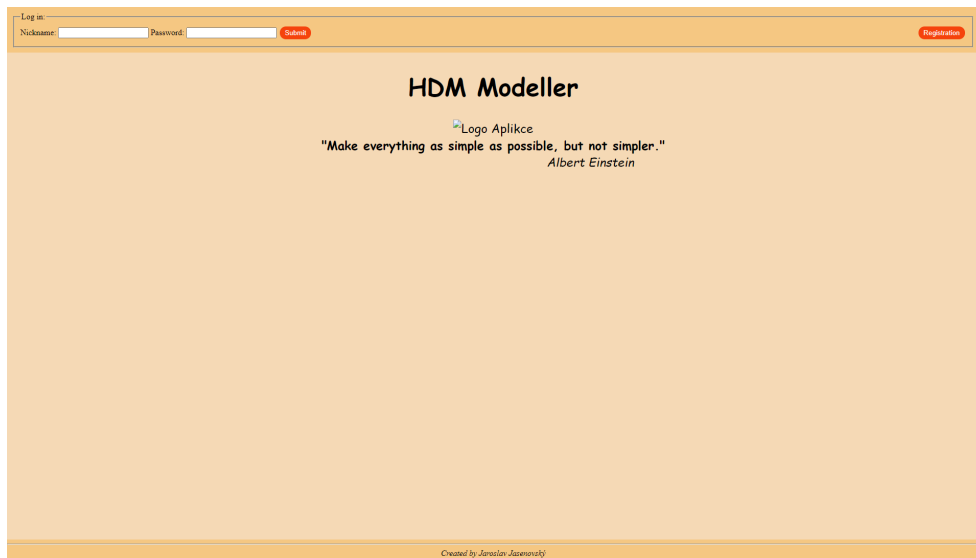
Pro uživatelské rozhraní je důležité, aby bylo zaměřeno na svou cílovou skupinu (uživatelé, kteří danou aplikaci využívají). Dále je také potřeba znát, na jakém zařízení se dané rozhraní bude zobrazovat. V poslední řadě je důležitý i smysl jednotlivých prvků a jejich logické rozložení, aby se i noví uživatelé zvládli v aplikaci rychle a intuitivně pohybovat. [3]

Vzhledem ke specifickému zaměření aplikace jsou i profily jejich budoucích uživatelů bez velkých odchylek dány. Zaměření je ryze technické a uživatelé jsou zdatní v používání moderních technologií. Také se jedná o studenty technických oborů a jejich kantory. U všech těchto uživatelů se dá očekávat, že ocení funkčnost více než nějaké zbytečné pozlátko. Z toho důvodu je navržen čistý vzhled, který nebude zbytečně křiklavý a nebude tolik dráždit oči. A samotné rozložení bude intuitivní a inspirováno klasickými aplikacemi, které daní uživatelé znají, a bude díky tomu samonávodné.

Hlavní funkcionalitou aplikace je tvoření modulů, které mohou být i značně rozsáhlé. Z tohoto důvodu se dá předpokládat, že uživatelé budou pracovat převážně u větších zobrazovacích zařízení, která budou orientována na šířku. Proto se vzhled orientuje převážně na ideální rozložení a zobrazení na těchto typech zařízení a dobrou čitelnost všech důležitých prvků, jež se na dané obrazovce nachází.

### 3.3.1 Návrh uživatelského rozhraní

V této části si ukážeme, jak vypadá převedení našich poznatků do praxe. Na jednotlivých modelových obrazovkách si jmenovitě ukážeme nejdůležitější prvky a hlavní myšlenku tohoto návrhu.

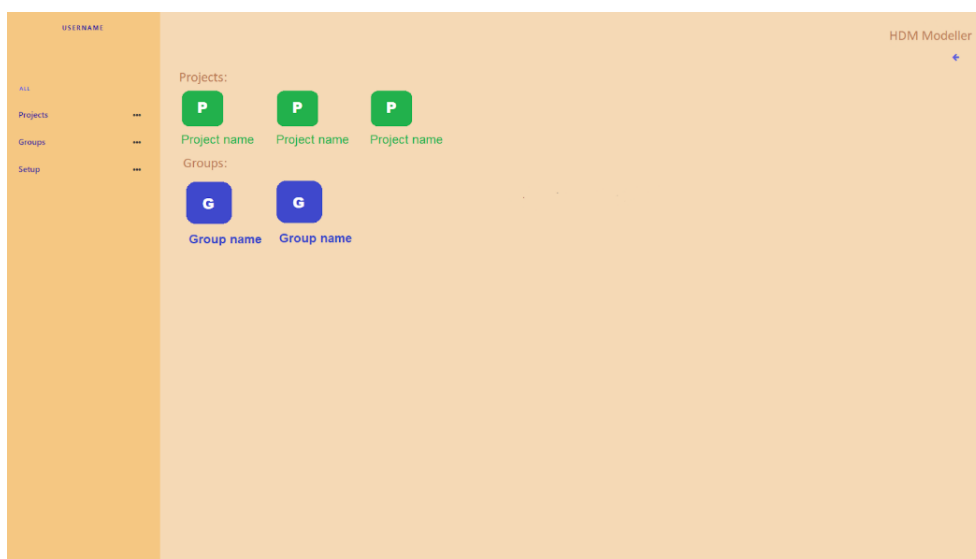


Obrázek 3.1: Návrh úvodní obrazovky pro přihlášení a registraci uživatele.

První, s čím uživatelé přijdou do styku, bude úvodní obrazovka s přihlašovací formulářem a možností se registrovat. Grafický styl poukazuje, jakým směrem se bude ubírat barevná paleta v celé aplikaci. Jak je vidět, jsou zvoleny jemné teplé barvy pro zmírnění únavy očí, a k tomu bylo použito dosti velké a kontrastní písmo pro snadnou čitelnost. Kromě samotných přihlašovacích formulářů je na této obrazovce obsaženo již pouze logo s názvem aplikace a krátké uvítací heslo, což vše poukazuje na směr, kterým se aplikace ubírá.

### 3. ANALÝZA A NÁVRH

---

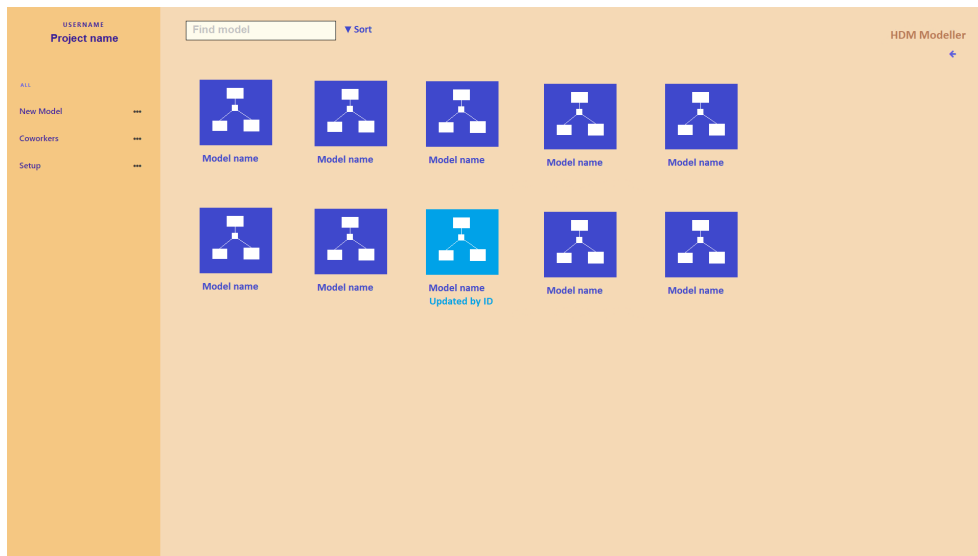


Obrázek 3.2: Návrh obrazovky uživatelského účtu, kde se nachází hlavní rozcestník pro další činnosti.

Po přihlášení se uživatel ocitne na obrazovce svého uživatelského účtu, kde se nachází lišta rozcestníku, a v případě, že uživatel již vytvořil nějaké projekty, tak posledních pár bude zobrazeno v rychlém výběru. Stejně tak, pokud je uživatel členem nebo správcem nějaké skupiny. Na liště jsou základní rozkládací položky, jako projekt či nastavení, a při rozbalení nabízejí další možnosti. Při procházení vlastních projektů a skupin je možnost třídít tyto projekty dle data či názvu a je možnost i vyhledávat dle názvu. Tato možnost usnadní a urychlí procházení mezi projekty (skupinami).



### 3.3. Analýza uživatelského rozhraní

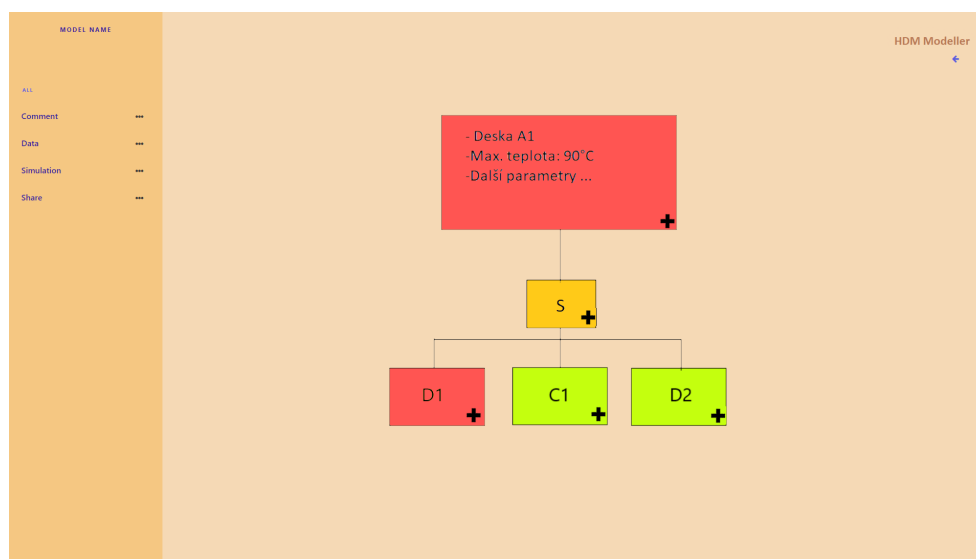


Obrázek 3.3: Návrh obrazovky zobrazující určitý projekt uživatele a obsahující modely, které do projektu náležejí.

Na obrazovce určitého projektu je opět panel rychlého výběru, na kterém se nachází možnost vytvořit nový model, ale také možnost přidat nebo odebrat spolupracovníka, kterého lze vyhledávat podle jména, příjmení či emailu. Dále je zde výpis všech modelů, které k projektu patří, opět s možností řazení a vyhledávání, a pokud některý ze spolupracovníků aktuálně pracuje na některém modelu, tak je odlišen.

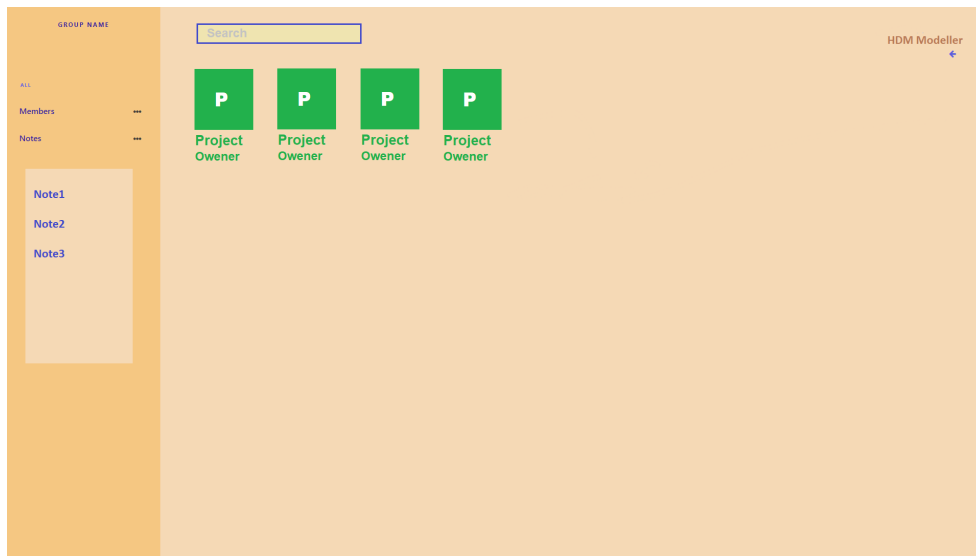
### 3. ANALÝZA A NÁVRH

---



Obrázek 3.4: Návrh obrazovky zobrazující rozpracovaný model. Na obrazovce se vždy zobrazuje pouze blok daného modelu, aby byla co nejsnazší a přehledná.

Při práci s modelem je součástí panelu rychlého výběru možnost odeslat náhled modelu, přidat komentář a popis. Pro lepší zobrazení se vždy z modelu zobrazuje pouze blok subsystému, ve kterém se nacházíme. Ke každé buňce (subsystému) lze přidat komentář, nastavit hodnoty jak běžné, tak kritické nebo přidat další potomky. Veškeré tyto operace budou řízeny skrze samotné buňky, a pokud bude mít některý subsystém hodnoty, které přesahují parametry celku, bude zvýrazněn červeně a s ním i cesta k částí, kde byly hodnoty překročeny. Rychlý výběr obsahuje také tlačítko simulace, kde lze nastavit okolní parametry, které mohou celý systém ovlivnit, a pokud někde bude opět překročena kritická hodnota, bude tato část i s celkovou cestou vlivu zvýrazněna. Zároveň se v logu simulace objeví přímý odkaz na subsystém, kde byly hodnoty překročeny.



Obrázek 3.5: Návrh obrazovky pro skupinu, kde se zobrazují všechny projekty skupiny a také je zde nástěnka pro poznámky.

Poslední z hlavních obrazovek je ta pro skupinu. Zde bude hlavní položkou rychlého výběru správa členů a dále tu také bude nástěnka. Budou zde zobrazeny projekty, jež sem přidali členové, a opět řazení a vyhledávání mezi nimi. Správce také vidí aktivitu na jednotlivých projektech a bude moci vlastníkům projektů nechávat osobní poznámky k jejich práci.

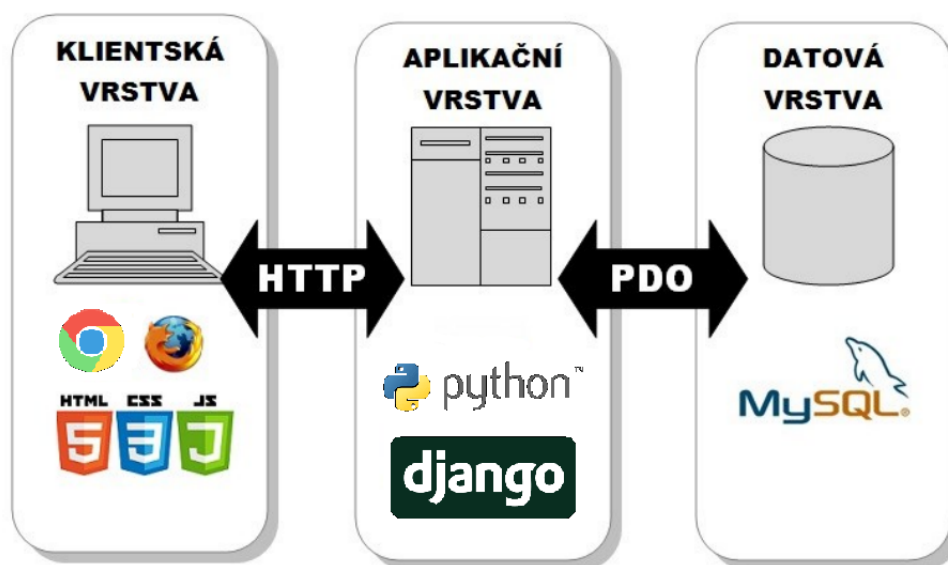
## 3.4 Analýza použitých technologií

Každý systém se skládá z několika částí a vzhledem k webovému rozhraní lze vyčlenit tyto základní složky:

**Frontend** je část aplikace z pohledu uživatele. Obsahuje tedy vše, co je vidět, a tvoří tak uživatelské rozhraní aplikace. Frontend běží na straně uživatele a je překládán v prohlížeči. Pro tuto část není velká možnost volby vzhledem k nutné podpoře prohlížečů, a proto je tato část navrhována v HTML5 a CSS3, které jsou aktuálními verzemi těchto technologií.[4]

**Backend** je mozkiem celé aplikace. Má za úkol zpracovávat veškeré příkazy, data a jejich administraci. Backend běží na straně serveru, a není tak omezen aplikační podporou na straně uživatele. V tomto případě je volba širší, ale vzhledem k potřebě analýzy dat a rychlosti jsem zvolil Python s frameworkem Django. Dalším velkým plus je i jeho jednoduchost, velká podpora a oblíbenost. Dále je jeho výhodou také skvělá práce s daty a jejich zpracování.[4]

**Databáze** je úložištěm veškerých dat, ať už se jedná o uživatele, jejich projekty či samotné modely. Vzhledem ke zjevným vztahům mezi veškerými objekty navrhuji tuto aplikaci pro relační databázi a vzhledem ke školnímu použití volím MySQL, aby nebyla třeba drahá licence od Oraculu či složité a také placené řešení od Microsoftu.[5]

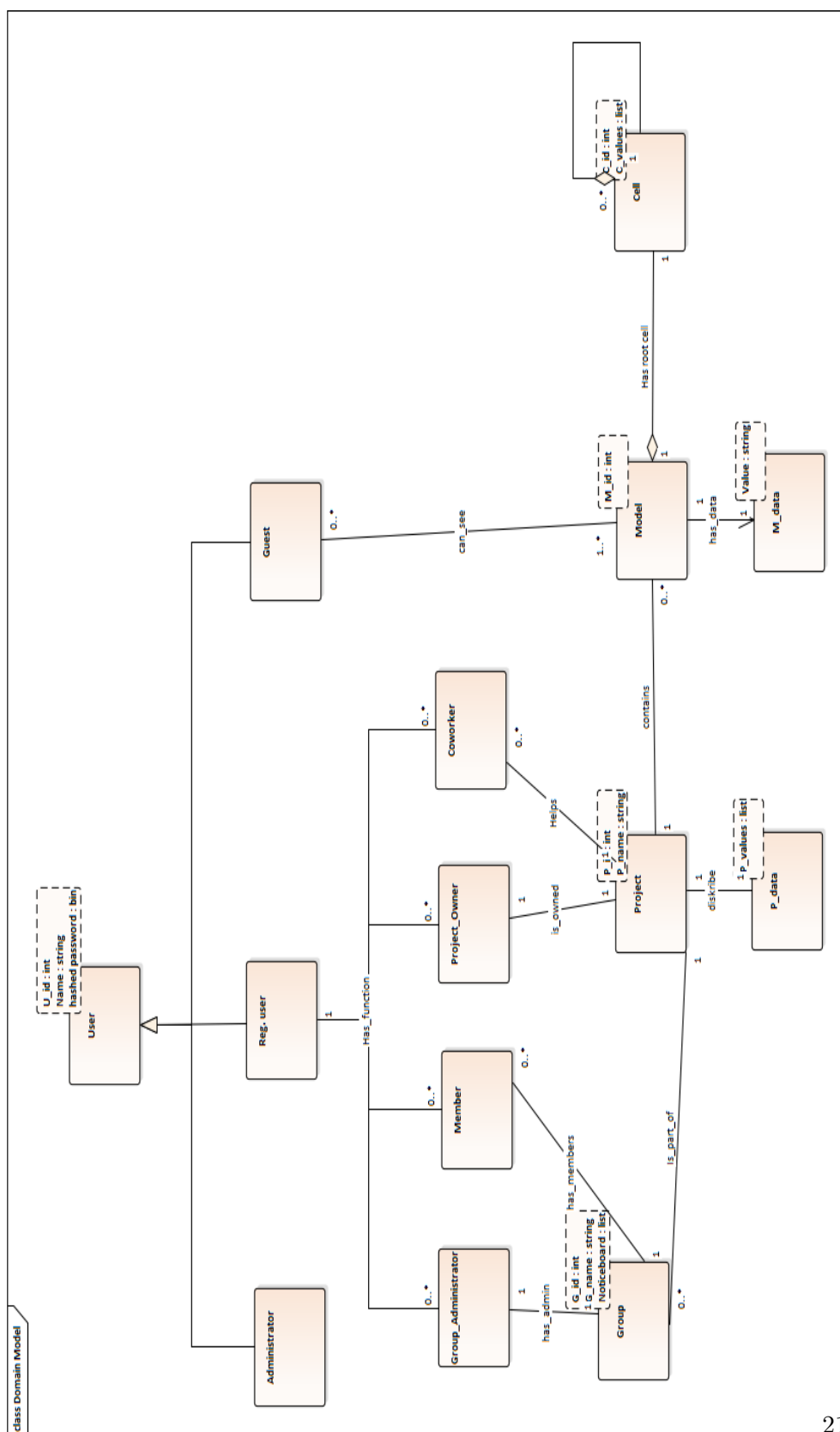


Obrázek 3.6: Vrstvy navrhovaného systému s použitými technologiemi.

Pro budoucí aplikaci se jedná pouze o doporučení a záleží na samotných vývojářích, které technologie pro svou práci upřednostňují. Například lze použít jiný typ databází, ať už placených, nebo verzi zdarma od Microsoftu, která je určena pro nekomerční využití. Dále je tu i široký výběr jazyků pro aplikační vrstvu, které jistě udělají stejnou službu, a záleží tedy hlavně na preferenci a zkušenosti vývojářů.

## 3.5 Relační návrh aplikace

Tento návrh je základem pro následné vytvoření databáze a vlastních tříd. V návrhu je důležité jednotlivé třídy dostatečně zapouzdřit, aby jedinou jejich vlastní závislostí byla jejich relace.



Obrázek 3.7: Relační model webové aplikace znázorňující vztahy jednotlivých objektů v modelu. Slouží pro vytvoření databáze a tříd, jež mají s danými objekty pracovat. Model byl vytvořen v aplikaci Enterprise Architect

### 3. ANALÝZA A NÁVRH

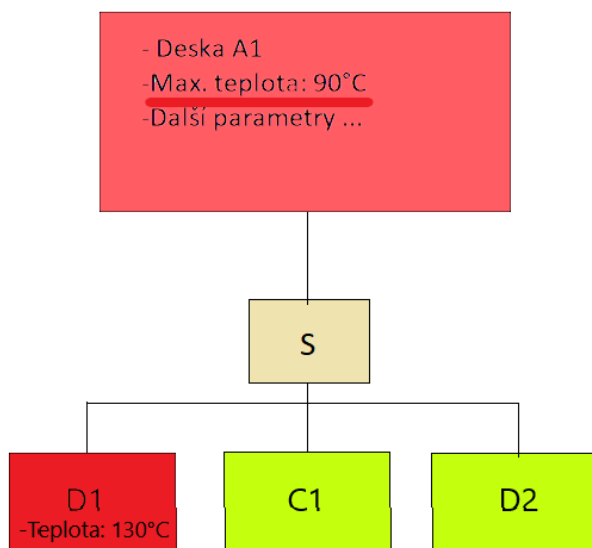
---

Většina vztahů je jednoduchých a stačí pro ně pouze jen vytvořit relace v databázi. Součástí aplikace jsou ovšem také netriviální vztahy, které si nyní lépe rozvedeme, aby bylo zřejmé jak fungují.

Každý účet bude spojený s emailovou adresou uživatele a každý důležitý proces vyvolá akci, která pošle informační email uživateli, jehož se to týká. Tyto procesy jsou například pozvání do skupiny, pozvání do projektu nebo snaha spolupracovníka smazat model. V nastavení bude mít uživatel možnost si nastavit, pokud nebude chtít pro nějaký proces email zasílat.

Další logikou aplikace je validace modelu. Každý model se skládá z jednotlivých hierarchicky srovnaných bloků. Každý blok se skládá z kořenové buňky, pravidlo podle něž se daný funkční celek rozpadá (nejčastěji sériové či paralelní). A následuje množina subsystémů, které jsou již listy, nebo jsou kořenem vlastního bloku. Tyto věci je třeba hlídat, a proto aplikace kontroluje jestli každý blok modelu obsahuje tyto části a také zda mají zadané vlastní parametry.

Poslední důležitou částí je zobrazení chyby a to jak při tvorbě modelu, tak při jeho simulaci. Při tvorbě se hlídá, aby si nějaké kritické hodnoty neodporovali. Například, když bude mít nějaký funkční celek zadáno, že může mít maximální teplotu 50°C a nějaká jeho součástka má parametr, že se zahřívá na teploty 150°C. Při simulaci se naopak nastaví simulované podmínky a to třeba velikost napětí či teploty. Aplikace následně vyhodnotí jestli mají dané podmínky vliv na nějaké součástky a pokud ano tak je označí a vyznačí i jejich vliv na následné funkční celky až třeba na celý systém.



Obrázek 3.8: Grafické znázornění kritických částí v daném bloku modelu. Lze krokovat až k původu dané chyby.

---

# Studie Proveditelnosti

„Studie proveditelnosti (Feasibility study) obsahuje systematicky uspořádané informace potřebné pro celkové vyhodnocení projektu. Má za úkol zhodnotit různé alternativy a posoudit uskutečnitelnost daného investičního projektu a poskytnout veškeré podklady pro investiční rozhodnutí. Je nástrojem pro posouzení návrhu projektu zejména z ekonomického a technického hlediska.“ [6]

Tato osnova je přizpůsobena potřebám daného projektu a vychází z doporučené struktury a stanov Ministerstva pro místní rozvoj.

1. Úvodní informace
2. Stručný popis podstaty projektu a jeho etap
3. Analýza trhu
4. Technické a technologické řešení projektu
5. Hodnocení efektivity a udržitelnosti projektu
6. Řízení rizik
7. Závěrečné hodnocení projektu

## 4.1 Souhrnný obsah struktury studie proveditelnosti

**Úvodní informace** obsahují zejména účel, za jakým je studie zpracována, a informace o zadavateli a zpracovateli.

**Stručný popis podstaty projektu a jeho etap** nachází se zde charakteristika projektu a jeho etap. Obsahuje zaměření projektu a přehled, jakými etapami projekt prochází a čím jsou specifické. Také nastiňuje ostatní významná specifika projektu.

**Analýza trhu** zahrnuje popis všech marketingových hledisek projektu, jako je konkurenční řešení nebo velikost trhu. Dále také obsahuje analýzu uživatelů a konkurenceschopnosti výstupu.

**Technické a technologické řešení projektu** zahrnuje veškeré technické a technologické aspekty projektu včetně zvolených technologií, výhody a nevýhody tohoto řešení a z toho vyplývající rizika.

**Hodnocení efektivity a udržitelnosti projektu** vyhodnocení projektu pomocí kritériálních ukazatelů, které budou předem stanoveny (např. časová úspora, ušetřené další technologie, ...)

**Řízení rizik** vymezení největších zdrojů rizika v projektu, jejich ohodnocení, stanovení opatření pro jejich mitigaci a scénáře jak jednat, pokud nastanou.

**Závěrečné hodnocení projektu** obsahuje závěr, který zahrnuje výsledné posouzení projektu a jeho realizovatelnosti. [7]

## 4.2 Úvodní informace

Tato studie je zpracována za účelem zhodnocení přínosů navrhované webové aplikace pro tvorbu HDM. Zadavatelem této studie je Ing. Martin Daňhel, Ph.D. jakožto vedoucí mé bakalářské práce a zástupce Českého vysokého učení technického v Praze a její Fakulty informačních technologií. Zpracovatelem je Jaroslav Jasenovský, tvůrce této bakalářské práce a samotného návrhu aplikace dle zadání vedoucího.

## 4.3 Stručný popis podstaty projektu a jeho etap

Tento projekt je specifický hlavně tím, že i když se snaží stavět na základech a principech, které již všichni známe, tak vyvíjí něco zcela nového, co tu ještě nikde není. Je tudíž těžké hledat zde srovnání, neboť přestože lze tyto modely vytvářet v mnoha různých aplikacích a mnohé jsou i postaveny na webovém rozhraní, žádná není zcela nativní. Z tohoto důvodu tu máme dva úhly pohledu. Je to něco zcela originálního, a tudíž bezkonkurenčního. Na druhou stranu se ovšem díky této specializaci zmenšuje potencionální trh, kde může být využívána.

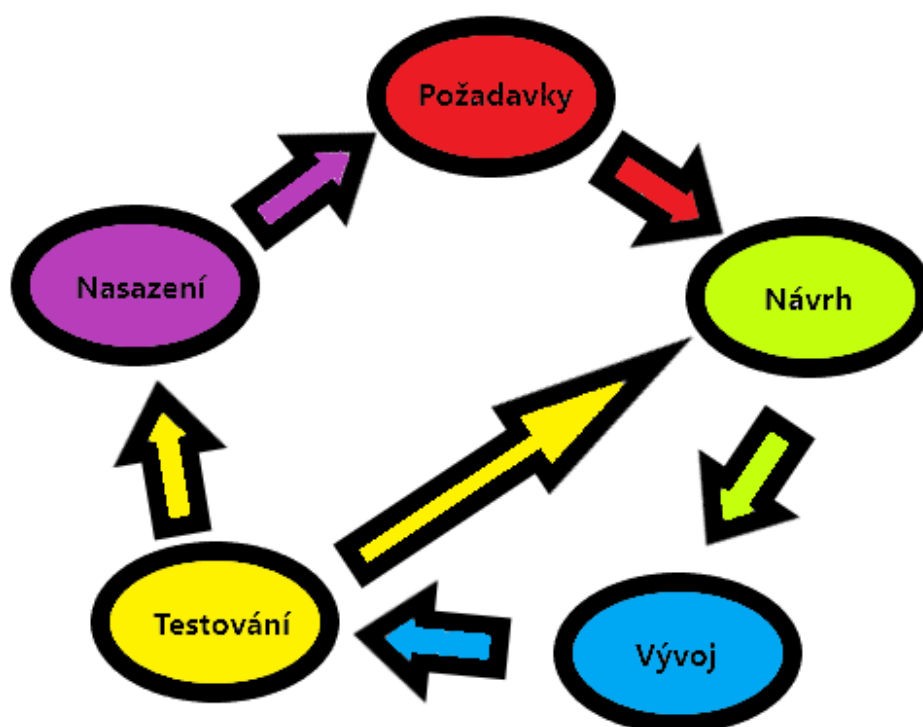
Samotné etapy tohoto projektu jsou děleny a specifikovány na:

1. **Analýza a návrh aplikace** obsahuje návrh rámcového řešení na základě požadavků. Obsahuje popis a ukázky řešení uživatelského rozhraní i samotné logiky aplikace a datové části. Tato etapa je součástí této bakalářské práce a je jedním z jejích výstupů.



2. **Vývoj modeláře.** V další etapě je cílem vyvinout samotný modelář, kde bude moci uživatel vytvářet příslušné modely. Tato etapa bude dělena ještě na další menší části:
  - a) Jednoduché modelování – Modelář pro vytváření modelu bez vyšší logiky.
  - b) Výpočet kritických hodnot – Model bude sám dopočítávat kritické hodnoty a zvýrazňovat kritické body v systému.
  - c) Validace modelu – Aplikace bude kontrolovat validitu v modelu, aby odpovídal hierarchické struktuře a obsahoval validní či kompletní hodnoty.
  
3. **Vývoj přihlašovacího rozhraní** obsahuje přihlašovací proces uživatelů a jejich přístup do vlastního rozhraní, kde budou mít uloženy vlastní projekty a modely. S tím souvisí i systém rolí a práv uživatelů.
  
4. **Vývoj skupinového rozhraní** je poslední část aplikace, která obsahuje skupiny a propojení uživatelů. V této části bude přidáno několik dalších rolí, díky kterým bude v aplikaci zcela podporována týmová práce.
  
5. **Údržba** je klasickou službou, kdy je aplikace udržována ve funkčním stavu a jsou zde opravovány případné chyby.

Veškeré tyto etapy, jakožto i celá aplikace fungují samozřejmě na principu vývojového kruhu. Aplikace se po každém vývoji testuje, a následně se analyzují další požadavky a chyby, které jsou opraveny. Tento proces probíhá opakovaně tak dlouho, dokud není část i celek v pořádku.



Obrázek 4.1: Životní cyklus vývoje systému, kde je nejdříve potřeba analyzovat požadavky. Následuje návrh aplikace a její vývoj, který je dále testován a konzultován. Pokud se objeví nějaké nové požadavky či problémy, tak se opět vrací do návrhu, jinak je daná část nasazena a jsou řešeny další požadavky.

### 4.4 Analýza trhu

V první řadě je důležité si určit, pro jaký trh je tato aplikace určena, a díky tomu pak lze najít hlavní konkurenční řešení, vůči kterému porovnáme navrhovanou aplikaci.

První cílovou skupinou jsou určitě školy a jejich studenti s technickým zaměřením. Při výuce, kde se studenti učí sestavovat systémy, je jistě potřeba ověřit správnost návrhu a zkontrolovat jejich spolehlivost; tudíž zde najdou všemožné modely uplatnění, neboť se používají i v běžné praxi, a tak je to skvělá průprava.

Ve školství ještě zůstaneme a vezmeme v potaz i samotné vyučující a zaměstnance, kteří při své vědecké práci navrhují již složitější systémy, kde si už s jednoduššími typy modelů často nevystačíme, a je tedy vhodné mít možnost pracovat s něčím, co zvládá i ty nejsložitější návrhy.

Odbytiště dále nalezneme i v praxi. Na trhu figuruje značné množství technologických firem, které navrhují systémy a subsystémy pro celou škálu strojů. Jedná se o automobilový a železniční průmysl, který u nás zaujímá velký podíl na trhu, ale i o spoustu jiných odvětví, ve kterých je potřeba, aby byly jejich systémy stabilní a spolehlivé.

Dále je třeba si specifikovat samotného uživatele. Jedná se o lidi, kteří se zabývají technologiemi, a jsou tedy technicky zdatní a práce na počítači by pro ně neměla být problém. Také spíše než samotné pozlátko ocení jednoduchost a funkčnost aplikace. Avšak usnadnění práce v podobě automatických výpočtů a nalezení kritických částí je jedním z hlavních důvodů, proč tyto aplikace používat, protože šetří čas i práci a minimalizuje pravděpodobnost chyby, která by mohla vést k vývoji nespolehlivého systému. V neposlední řadě je zde důležitým prvkem potřeba skupinové práce mezi uživateli, neboť jak ve škole, tak ve firmě je třeba propojení lidí a týmů, díky čemuž se práce usnadní, urychlí, a opět dochází k větší kontrole, což vede k lepším výsledkům.

Je samozřejmé si uvědomit, že tyto firmy mají často již zavedené aplikace, ale každá moderní firma musí jednou za čas projít aplikační obměnou. A vzhledem k tomu, že cílovou skupinou této aplikace jsou i školy, které budují nové generace pracovníků, tak se s touto aplikací setkají již na školách, a díky tomu máme velkou konkurenční výhodu, protože lidé dávají přednost věcem, které už znají. A co se týče výhod oproti konkurenci ve školách, tak je to jistě zaměření na komunikaci a interakci mezi studenty a vyučujícími, což usnadní jejich práci, a není třeba využívat další aplikace třetích stran. Dále je to i samotná automatizovaná kontrola a výpočet, což usnadní práci, a díky využití hierarchického modelu, který lze aplikovat na řešení triviálních systémů, ale i na ty nejsložitější typy problémů, se jedná o aplikaci, která “poroste” s uživatelem.

V poslední části se podíváme na samotnou konkurenci, co nabízí a jaké jsou její výhody a nevýhody.

- Empirical Modelling – Dependency Modeling Tool [8]
  - Závislosti jsou psány pomocí skriptů.
  - Pracuje pomocí stavů.
  - Program sám generuje stavový graf.
  - Výhody:
    - \* Automatická generace grafu.
    - \* Jednoduchý skriptovací kód (a is b + c, b is d, ...)
  - Nevýhody:
    - \* Podporuje jen základní typy spolehlivostních modelů.
    - \* Při aplikaci na složitější problém začne být skriptovací kód i generované grafy nepřehledné a složité.
    - \* Zhoršená možnost spolupráce.



- \* Není to nativní aplikace pro práci se spolehlivostními modely.
- Sharpe [10]
  - Zaběhlý program s širokou paletou použití.
  - Lze v něm modelovat různé typy modelů.
  - Výhody:
    - \* Volnost při modelování.
    - \* Zavedený program.
    - \* Čitelné zobrazení modelů.
  - Nevýhody:
    - \* Nemá nativní podporu HDM.
    - \* Ne příliš intuitivní prostředí pro různé typy modelů.
    - \* Chybí kontrola validace.

## 4.5 Technické a technologické řešení projektu

Podrobný popis technických parametrů aplikace najdeme v samotném návrhu. Proto se zde budu věnovat hlavně klíčovým prvkům řešení a jejich výhodám či nevýhodám a s tím spojeným případným rizikům.

První důležitou vlastností je, že se jedná o webovou aplikaci, což s sebou přináší spoustu specifik. Začneme výhodami, mezi něž patří rozhodně nižší nároky na uživatelské pracovní stanice, protože veškeré složité výpočty budou prováděny na straně serveru. Další výhodou je zajisté i snadný přístup odkudkoliv, kde má uživatel přístup k internetu, což podporuje práci studentů z domova, ale i možnost pro zaměstnavatele zajistit zaměstnancům možnost homeoffice.

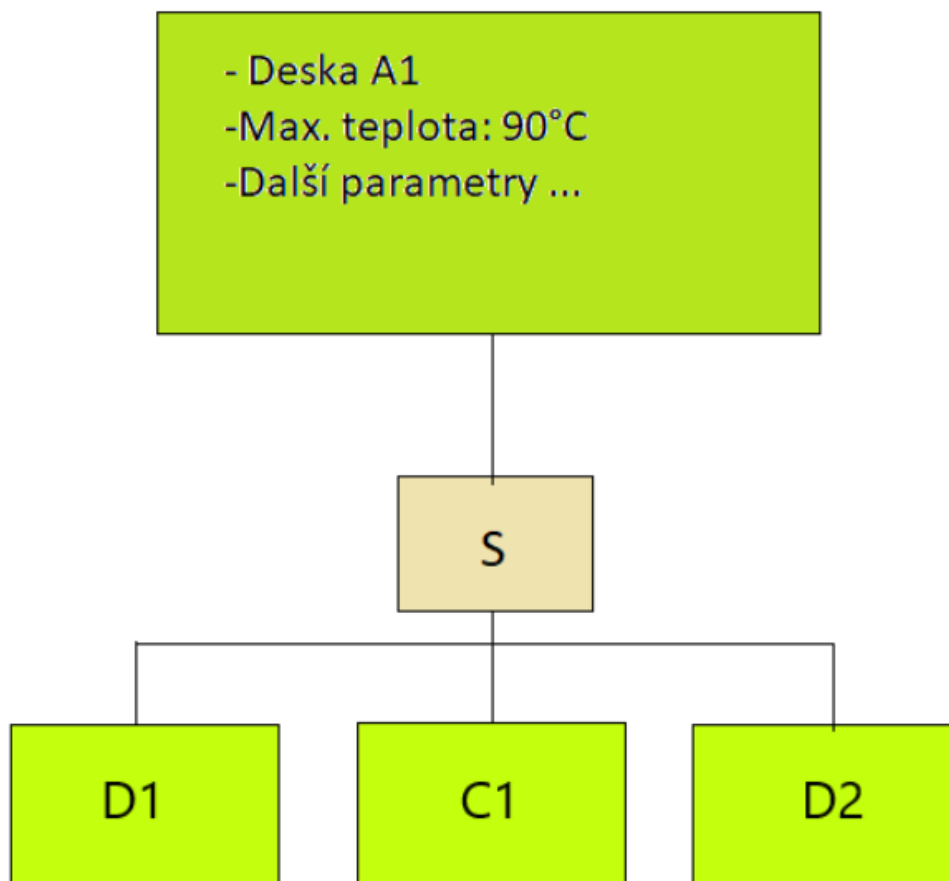
Mezi nevýhody tohoto rozhraní může být nutnost přístupu k internetu, což už v dnešní době bývá samozřejmostí, avšak stále je tu riziko výpadku internetu či jiná síťová omezení. Další velkou nevýhodou je i nutnost investice do infrastruktury, neboť jsme sice snížili nároky na pracovní stanice, ale tím jsme zvýšili nároky na serverovou stranu, kde pokud by firma investovala do vlastní infrastruktury, bude se jednat o větší jednorázový výdaj. Ovšem je tu i možnost cloudového řešení, kdy je serverová část pronajata od třetí strany a je zajištěn požadovaný výkon a vlastnosti.

Druhou důležitou vlastností je její specializace, díky čemuž může uživateli maximálně práci usnadnit. A to převážně díky automatickému výpočtu pravděpodobností, zvýrazněním kritických hodnot a jejich propagace v hierarchickém modelu, což z pohledu odspodu ukáže jejich celkový vliv na celistvý systém, a zároveň z pohledu ze shora je možné dohledat kritické části do jejich základního prvku. Poslední usnadnění při modelování je i verifikace modelu, díky tomu je model korektní, a tudíž mohou zůstat korektní i výpočty. Toho je

#### 4. STUDIE PROVEDITELNOSTI

---

docíleno díky stromové struktuře modelu, kdy je možné nějaký systém rozdělit na jeho subsystémy až do základní části (listu).



Obrázek 4.3: Ukázka bloku v HDM, který je je zapojen sériově a skládá se z jednoduchých součástek (cívka, dioda).

Nevýhodou tohoto řešení je hlavně zmenšení možného trhu a omezení uživatele, který může pracovat pouze v tomto specifikovaném modelu. Také díky této restrikcí hrozí, že potenciální uživatelé z nechuti učit se něco nového a z toho vyplývajícího strachu ze složitosti, dají přednost jinému řešení.

Poslední vlastností je její zaměření na skupinovou a týmovou práci, neboť tento model krásně sedí na běžná vývojová pracoviště technických firem, ale také práci studentů na školních projektech během výuky. Je tu důležitá práce správce skupiny, který tak může kontrolovat práci členů dané skupiny, ale i možnost spolupráce několika uživatelů na jednom projektu. V neposlední řadě také možnost poslat určitý model člověku, který není registrovaným uživatelem.

vatelem a má v něm pouze možnost náhledu, a je omezen pouze na daný model.

Tento návrh má nevýhodu v trochu složitější rozhraní pro uživatele a hlavně složitější samotný vývoj, neboť tu vzniká velké množství závislostí mezi různými objekty.

Co se týče technologií, tak u frontendu není u webového rozhraní jiná možnost volby. Co se týče backendu, tam je sice široký výběr, ale vzhledem k velké oblíbenosti Pythonu, jeho podpoře a hlavně skvělé schopnosti zpracovávat data a rychlý běh jeho výpočtu si myslím, že to je dobrá volba a jen stěží hledat nevýhody.

Poslední strana je databáze. Tady bylo zvoleno úsporné řešení, hlavně kvůli danému školnímu nasazení, a celkově s MySQL nebývá moc často problém, ale pořád je tu riziko menší podpory vyplývající z této volby.

## 4.6 Hodnocení efektivity a udržitelnosti projektu

Hodnocení efektivity bude ukázáno na klíčových procesech tvorby spolehlivostních modelů. Proces bude vždy popsán tabulkou, pak porovnán z hlediska stávajících možností (bude-li to možné) a z hlediska navrhovaného systému. Při tomto srovnání bude hlavními kritérii časová úspora, automatizace procesů a nutnost využití dalších technologií. Procesy jsou modelovány pomocí BPMN a BORM. (Obrázky modelů procesů budou obsaženy v příloze pro plné zobrazení ve formátu PNG)

Sledovanými procesy jsou:

- Tvorba modelu
- Spolupráce na modelu
- Správa skupiny
- Náhled modelu

### 4.6.1 Tvorba modelu

Uživatel chce vytvořit validní model navrhovaného systému a spočítat spolehlivost daného systému.

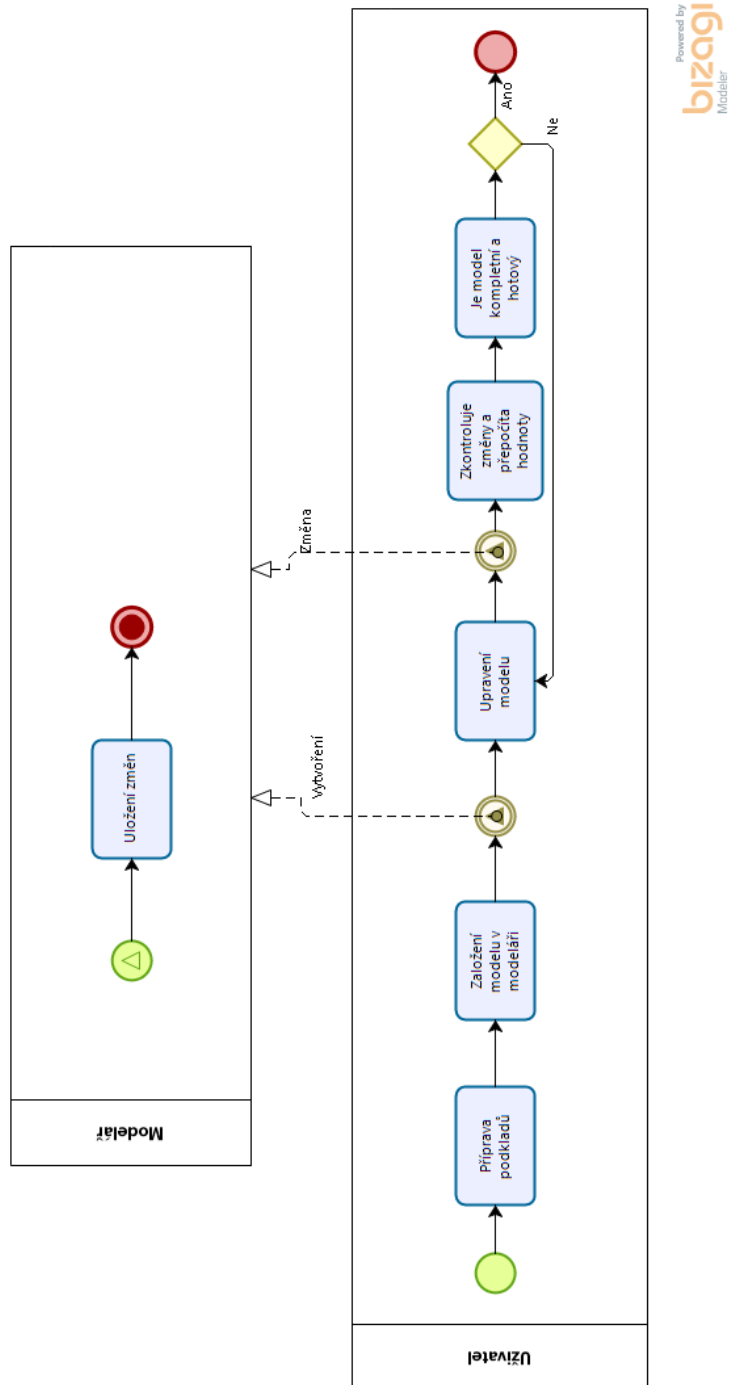
Aktuální řešení:

Tabulka 4.1: Popisná tabulka procesu tvorby modelu. Zde je popsáno aktuální řešení, kde není nativní aplikace pro HDM.

ID	Old01
Strategické cíle	Snadné a intuitivní vytvoření HDM modelu a jeho snadná validace.
Produkt/slужba	Vytvořený HDM model
Specifikace procesu	Vlastník modelu, který si připravil podklady pro daný model, si na své pracovní stanici otevře modelář, v němž začne kompletovat model pro daný problém. V průběhu kontroluje korektnost svých operací a model ručně validuje, dokud není jeho finální podoba korektní.
Vlastník procesu	Vlastník modelu
Oblast zlepšení / problémy	Automatická validace modelu. Podpůrné prostředky pro tvorbu modelu HDM. Usnadnění vřazaní přístupu k projektu a datům.
Metriky	Doba validace modelu, počet nekorektních operací v modelu, počet pracovních stanic.
Startovací událost	Založení modelu HDM.
Podmínky	Podklady pro model, korektní data a hodnoty



#### 4.6. Hodnocení efektivity a udržitelnosti projektu



Obrázek 4.4: BPMN model tvorby HDM v aktuálním řešení, kdy neexistuje aplikace podporující HDM. Model byl vyvíjen v aplikaci Bizagi modeler.

#### 4. STUDIE PROVEDITELNOSTI

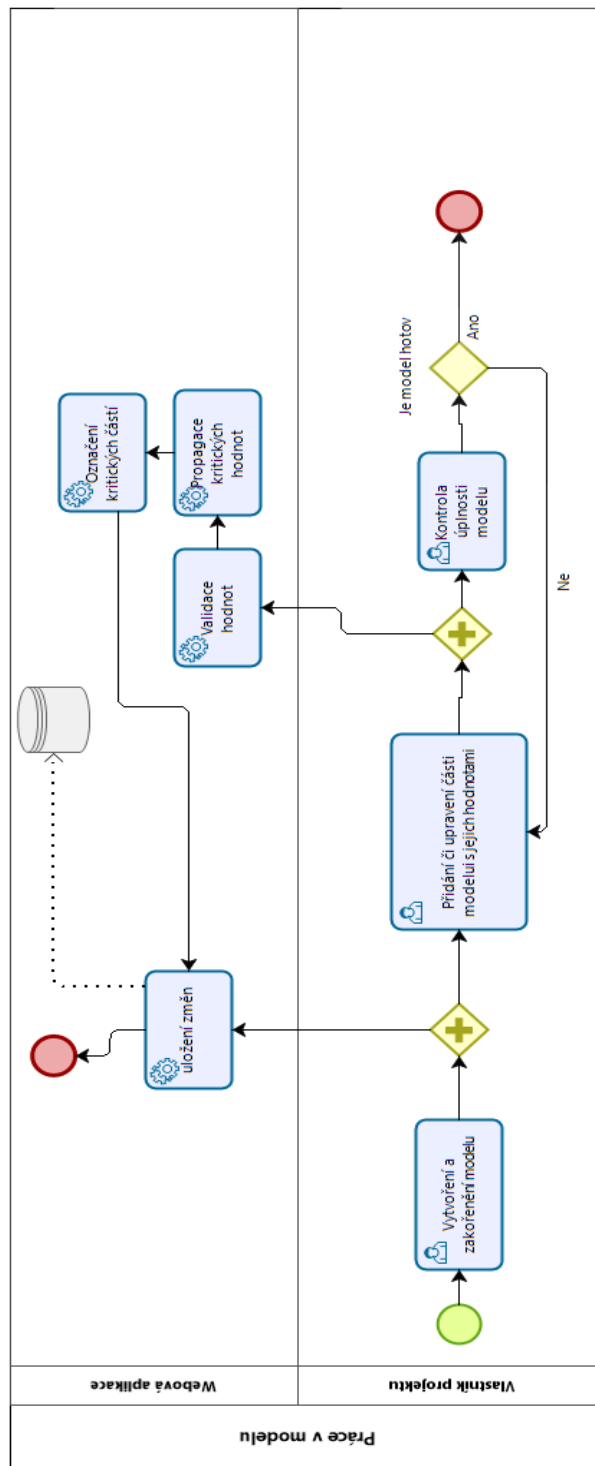
---

Navrhované řešení:

Tabulka 4.2: Popisná tabulka procesu tvorby modelu. Zde je popsáno navrhované řešení webové aplikace.

ID	New01
Strategické cíle	Snadné a intuitivní vytvoření HDM modelu a jeho snadná validace.
Produkt/služba	Vytvořený HDM model
Specifikace procesu	Vlastník na svém účtu ve WA zvolí či založí projekt, v němž chce daný model vytvořit. Zde pak vytvoří daný model, ve kterém jsou kritické hodnoty hierarchicky propagovány a validovány, a jsou v něm automaticky vyznačena kritická místa. Každý krok je automaticky ukládán v databázi a lze se vrátit zpět. Model je přístupný z jakéhokoliv vhodného zařízení po přihlášení do WA.
Vlastník procesu	Vlastník modelu
Oblast zlepšení / problémy	Zlepšena možnost přístupu, přidána automatická validace, systémová omezení dle specifikace HDM modelu (lze vytvořit pouze korektní modely).
Startovací událost	Kliknutí na tlačítko vytvořit nový model.
Podmínky	Podklady pro model, být registrován ve webové aplikaci, mít přístup k internetu.

#### 4.6. Hodnocení efektivity a udržitelnosti projektu



Obrázek 4.5: BPMN model tvorby HDM v navrhovaném řešení určeného pro HDM. Model byl vyvíjen v aplikaci Bizagi modeler.

#### 4. STUDIE PROVEDITELNOSTI

---

*Hodnocení procesu:* Je vidět, že složité části procesu, jako například validace a výpočet spolehlivosti, byly automatizovány a přesunuty na stranu aplikace. Díky tomu se samotná práce usnadní, minimalizuje se pravděpodobnost chyby (lidský element), a díky tomu se urychlí.

##### 4.6.2 Spolupráce na modelu

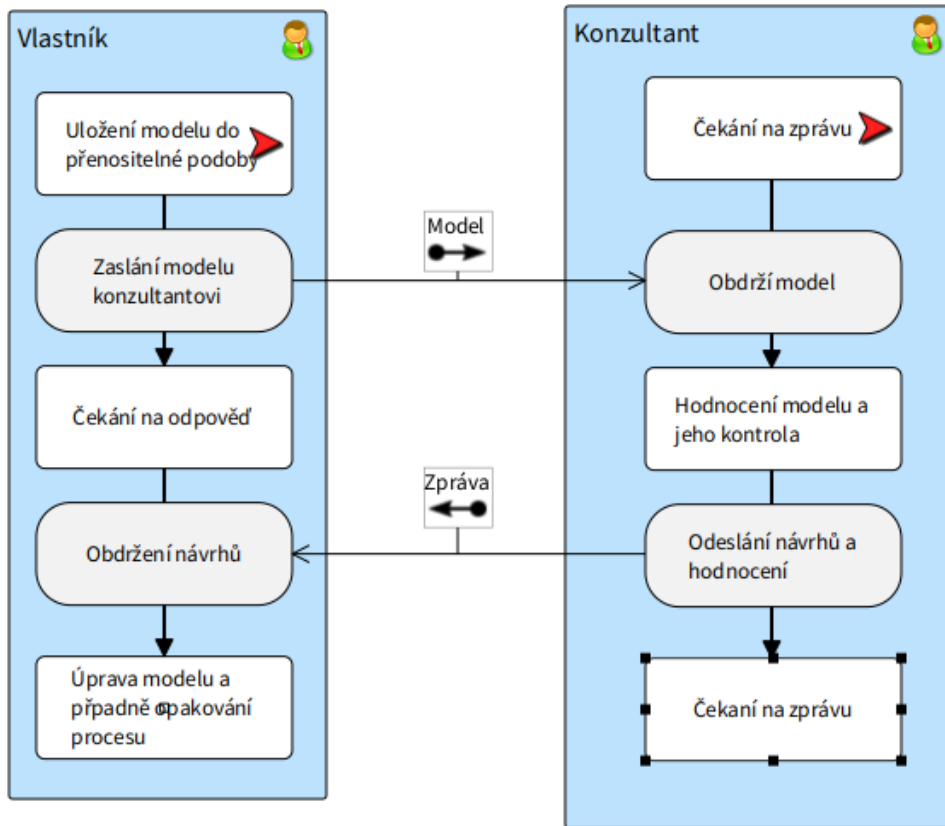
Při vývoji složitějších systému či skupinové práci ve škole je občas potřeba, aby více uživatelů spolupracovalo na vývoji jednoho modelu.

Aktuální řešení:

Tabulka 4.3: Popisná tabulka procesu spolupráce při tvorbě modelu. Zde je popsáno navrhované řešení webové aplikace.

ID	Old02
Strategické cíle	Snadná spolupráce a schvalování modelu.
Produkt/slужba	Konzultovaný model HDM
Specifikace procesu	Vlastník uloží model do nějakého přenositelného formátu. Ten následně zašle konzultantovi emailem. Konzultant u formátu v podobě obrázku zasílá své výhrady písemně nebo, v případě že dostane soubor v upravitelné podobě, ho modifikuje a opět pošle zpět s případným komentářem vlastníkovi. Popřípadě ukládání změn v nějakém verzovacím systému (git, SVN,...)
Vlastník procesu	Vlastník modelu
Oblast zlepšení / problémy	Snadný přístup k modelu odkudkoliv. zobrazení modelu v plném rozsahu, možnost kolegů do modelu zasahovat
Metriky	Rychlost sdílení modelu, doba propagace změn či návrhů stran sdílení
Startovací událost	Uložení modelu do přenositelného souboru (IMG, PDF, uložený projekt modeláře).
Podmínky	PDF/IMG reader, email/mailbox, modelář podporující daný typ souboru, verzovací systém

#### 4.6. Hodnocení efektivity a udržitelnosti projektu



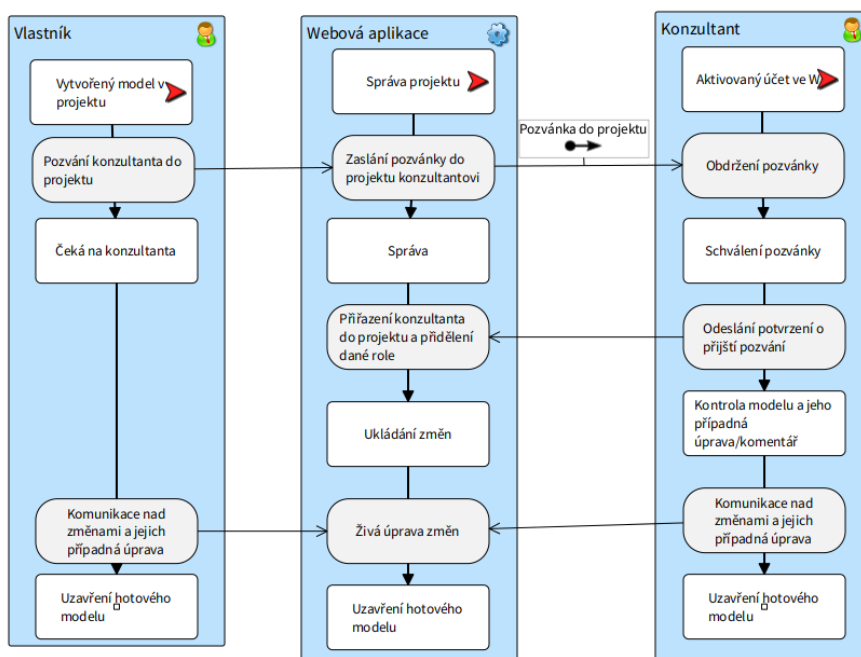
Obrázek 4.6: BORM model procesu sdílení modelu. V aktuálním stavu je třeba používat aplikace třetích stran jako například verzovací nebo zasílací služby. Model byl vyvíjen v aplikaci Openponk.

#### 4. STUDIE PROVEDITELNOSTI

Navrhované řešení:

Tabulka 4.4: Popisná tabulka procesu tvorby modelu. Zde je popsáno navrhované řešení webové aplikace.

ID	New02
Strategické cíle	Snadná spolupráce a schvalování modelu.
Produkt/služba	Konzultovaný model HDM
Specifikace procesu	Při potřebě spolupráce s konzultantem lze přidat do projektu spolupracovníka, který má práva modely v projektu upravovat (mazání podléhá schválení vlastníka).
Vlastník procesu	Vlastník modelu
Oblast zlepšení / problémy	Snadné sdílení a kompletní náhled na model. Možnost snadné spolupráce na modelu a okamžitá propagace změn obou stran.
Startovací událost	Kliknutí na tlačítko přidat spolupracovníka.
Podmínky	Mít model, kontakt na konzultanta.



Obrázek 4.7: BORM model procesu sdílení modelu. V navrhovaném řešení lze celý proces učinit v rámci navrhované webové aplikace. Model byl vyvíjen v aplikaci Openponk.

## 4.6. Hodnocení efektivity a udržitelnosti projektu

*Hodnocení procesu:* Po přidání kolegy jako spolupracovníka jsou změny z obou stran okamžitě propagovány a mohou na něm spolupracovat všichni najednou. Díky tomu dochází k časové úspoře. Další výhodou je, že vše probíhá pouze ve webové aplikaci, a není tak třeba využívat další aplikace a systémy pro jednu činnost.

### 4.6.3 Správa skupiny

Pokud je na nějakém projektu pracovní tým nebo ve škole v rámci výuky jedna třída, je vhodné tuto jednotku udržovat v rámci nějaké skupiny, kde má vůdčí osoba (teamleader, vyučující, ...) přehled o práci celé skupiny.

Aktuální řešení:

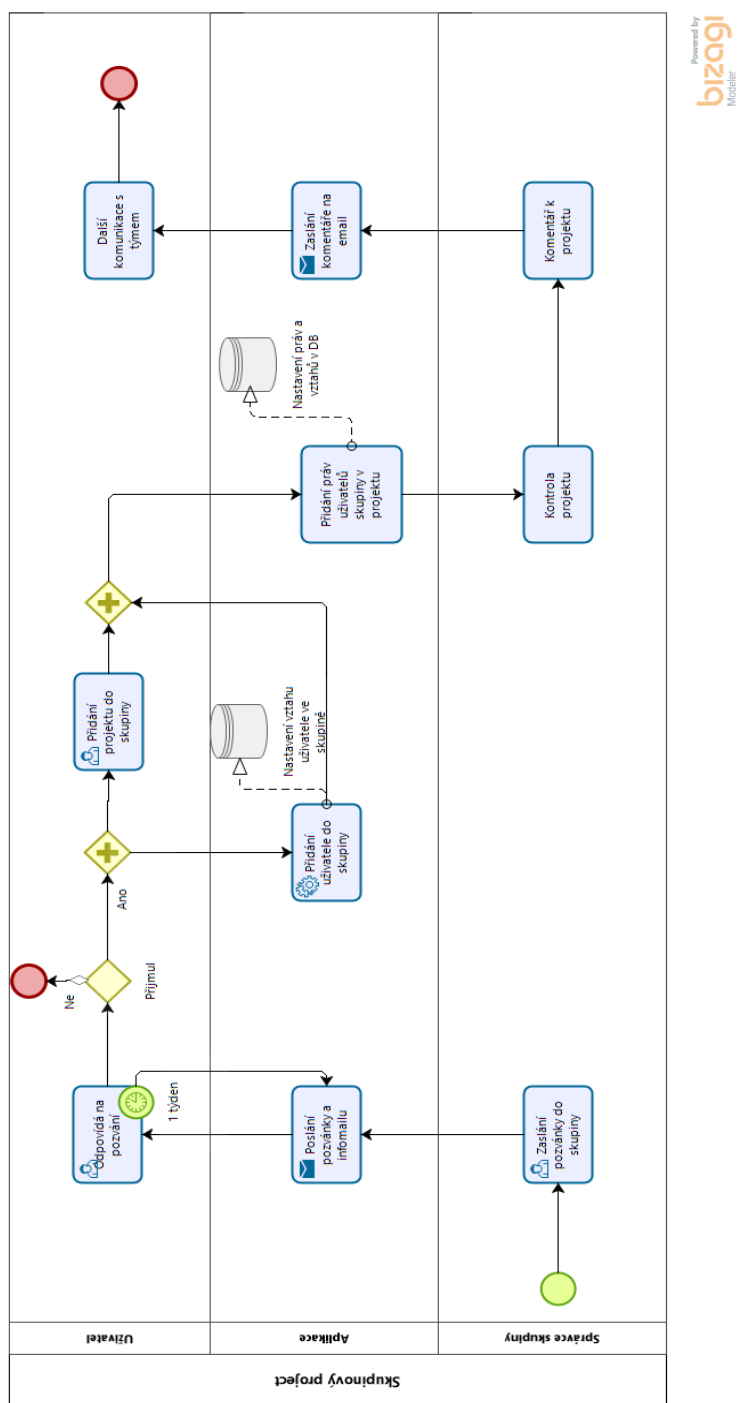
Lze řešit podobně jako v případě spolupráce, kdy je využit nějaký verzovací systém, či jsou výsledky zasílány průběžně.

Navrhované řešení:

Tabulka 4.5: Popisná tabulka procesu správy skupiny. Zde je popsáno navrhované řešení webové aplikace.

ID	New03
Strategické cíle	Usnadnění spolupráce uživatelů na větším projektu, kontrola nad prací členů skupiny.
Specifikace procesu	Registrovaný uživatel založí skupinu a stane se jejím správcem. Pozve další uživatele. Pozvání uživatelé přijmou pozvání a stanou se členy. Členové sdílí projekty ve skupině.
Vlastník procesu	Správce skupiny
Oblast zlepšení / problémy	Usnadnit a umožnit práci ve skupině.
Startovací událost	Založení skupiny
Podmínky	Znát uživatele, které chcete pozvat do skupiny. Přijmout pozvánku do skupiny.

#### 4. STUDIE PROVEDITELNOSTI



Obrázek 4.8: BPMN model procesu správy skupiny. Obrázek znázorňuje navržené řešení, kde je funkce podporována přímo v rámci webové aplikace. Model byl vytvořen v aplikaci Bizagi modeler.



## 4.6. Hodnocení efektivity a udržitelnosti projektu

*Hodnocení procesu:* Hlavním plusem navrhovaného řešení je snadné umožnění práce se skupinami a hlavně možnost vše dělat pouze v rámci navrhované aplikace, což velice usnadní práci.

### 4.6.4 Náhled modelu:

Někdy vyvstane potřeba poslat někomu pouze náhled modelu, aby ho mohl třeba schválit, či pro jiné účely.

Aktuální řešení:

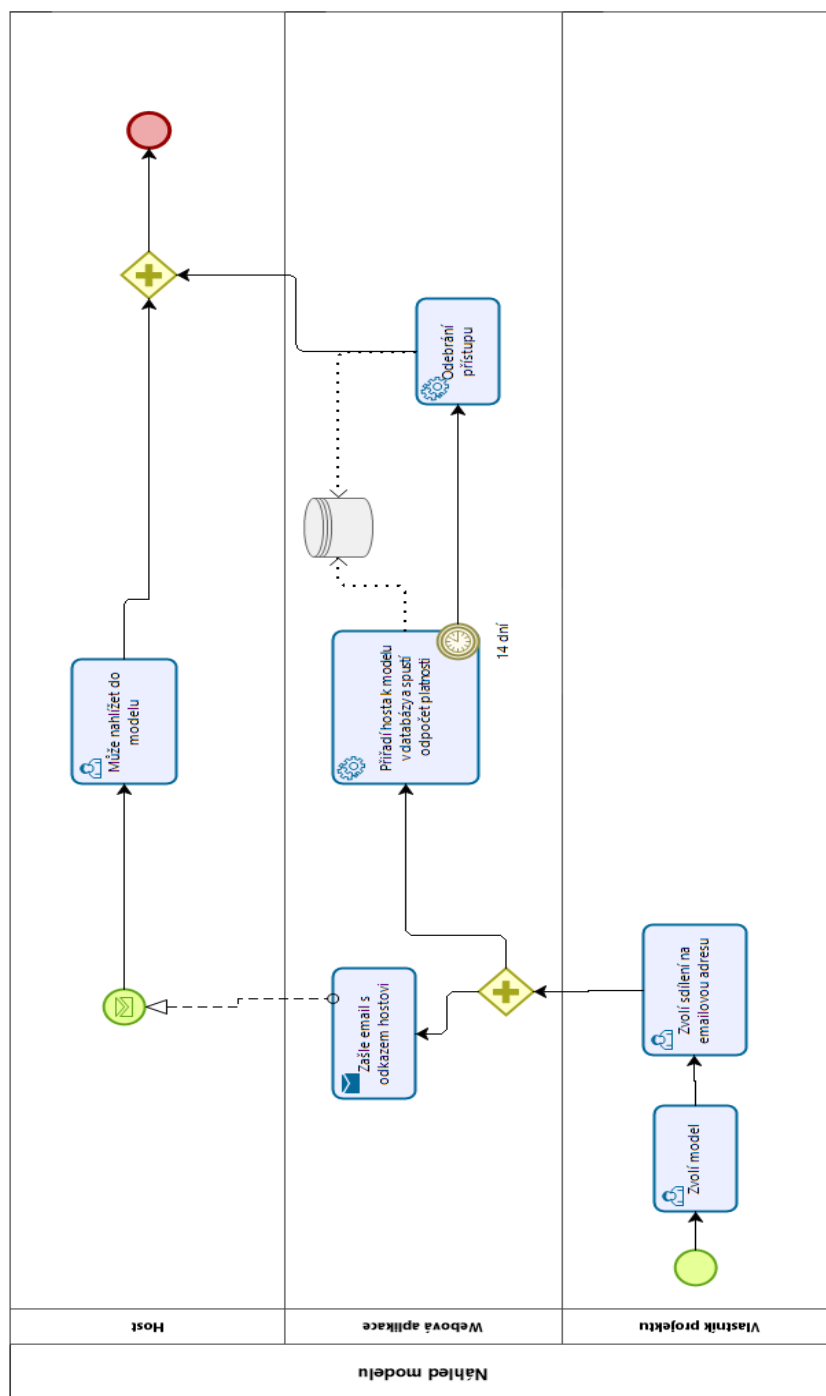
Opět řešeno podobně jako v případě spolupráce, kdy je zaslán emailem či jiným způsobem obrázek nebo přenositelná kopie daného modelu.

Navrhované řešení:

Tabulka 4.6: Popisná tabulka procesu náhledu modelu. Zde je popsáno navrhované řešení webové aplikace, kde je možné zaslat náhled i uživatelům, jenž nejsou registrovaní ve webové aplikaci.

ID	New04
Strategické cíle	Snadné sdílení náhledu modelu
Specifikace procesu	V případě pouhého náhledu lze sdílet pouze model za pomoci linku, který je zaslán na email, a umožňuje kompletní náhled na model s právy read-only (konzultant nemusí mít účet ve WA).
Vlastník procesu	Vlastník projektu
Oblast zlepšení / problémy	Snadné sdílení a kompletní náhled na model.
Startovací událost	Kliknutí na tlačítko sdílet.
Podmínky	Mít model a kontakt na konzultanta.

#### 4. STUDIE PROVEDITELNOSTI



Obrázek 4.9: BPMN model procesu zaslání náhledu na model. Navrhované řešení umožňuje sdílení pomocí odkazu, díky kterému získá daný host přístup read-only. Model byl vytvořen v aplikaci Bizagi modeler.

*Hodnocení procesu:* Celý proces je opět řešen pouze v navrhované aplikaci. Případné změny jsou stále propagovány do náhledu. Pokud vyvstane potřeba ukončit náhled, lze ho okamžitě odebrat nebo automaticky vyprší po 14 dnech a pro další náhled ho bude třeba obnovit. Narozdíl od obrázku muže v aplikaci při náhledu konzultant vidět celý model v plných detailech a nepotřebuje pro to žádnou aplikaci třetích stran.

#### 4.6.5 SWOT analýza

Tato analýza hodnotí silné a slabé stránky projektu z interního a externího pohledu. Má tak odhalit konkurenční výhodu i slabiny, na kterých je potřeba pracovat, ale také příležitosti a hrozby, které když dokážeme vhodně identifikovat a využít jich, popřípadě se jich vyvarovat, tak to může značně ovlivnit úspěch projektu.

Tabulka 4.7: Tabulka SWOT analýzy projektu, kde jsou zhodnoceny vnitřní i vnější silné a slabé stránky projektu.

Vliv	Pozitivní	Negativní
Vnitřní	<p>Silné stránky</p> <ul style="list-style-type: none"> <li>• Unikátnost projektu</li> <li>• Motivace</li> <li>• Spolupráce se školou</li> <li>• Schopnost přizpůsobit produkt potřebám trhu</li> </ul>	<p>Slabé stránky:</p> <ul style="list-style-type: none"> <li>• Zkušenosti</li> <li>• Neznámé jméno na trhu</li> </ul>
Vnější	<p>Příležitosti:</p> <ul style="list-style-type: none"> <li>• Možnost využití pro potřeby univerzity</li> <li>• Neexistující podobné řešení na trhu</li> </ul>	<p>Hrozby:</p> <ul style="list-style-type: none"> <li>• Zákaznické preference</li> <li>• Ukončení projektu</li> <li>• Duševní vlastnictví</li> </ul>

## 4.7 Řízení rizik

Pro co nejvíce bezproblémový průběh projektu je důležité si analyzovat veškerá možná rizika, která mohou nastat, a ohodnotit jejich pravděpodobnost a dopad na projekt. Následně u kritických rizik naplánujeme jejich řízení v po-

době určení vlastníka, jejich mitigace a plán zvládnutí daných rizik v případě, že nastanou. [11]

### 4.7.1 Analýza rizik

Nejdříve je důležitá pokusit se určit veškerá reálná rizika projektu. Po jejich nalezení si za pomoci matice pravděpodobnosti a dopadu určíme jejich závažnost a díky tomu můžeme rozhodnout, která rizika lze akceptovat a jaká je třeba dále řídit. Každé riziko má svůj popis, pravděpodobnost, že nastane (ta se může v průběhu projektu měnit) a důsledek.[11] Popis rizik:

- Nepřijetí aplikace - R1
  - I přesto, že je vyvíjená aplikace inovativní, má většina firem své zavedené postupy a bývá těžké do takto rozjetých firem zavádět nové systémy a metodiky, a to hlavně ze strany zaměstnanců, pro které to na první pohled znamená vypadnutí ze zaběhlých principů a učení se nových věcí.
  - Vyšší pravděpodobnost
  - Velký dopad
- Syndrom vyhoření - R2
  - Známá skutečnost, kdy po rozběhnutí projektu opadne prvotní entuziasmus, a s postupem času narůstá nechuť k dané práci, a jediné, co to může zachránit, je buď nový impuls, nebo rovnou celý osobní restart.
  - Malá pravděpodobnost
  - Vyšší dopad
- Zkušenosti - R3
  - Mé osobní zkušenosti jsou převážně získávány ve škole, kde pracuji na semestrálních pracech a podobných projektech. Z vlastní iniciativy se jedná jen o malé projekty, a tak vývoj něčeho v tomto měřítku bude zcela nová zkušenost.
  - Vysoká pravděpodobnost
  - Vyšší dopad
- Nemoc či zranění - R4
  - Nemoc ani zranění si nevybírají a rizika jsou všude kolem nás. A vzhledem k tomu, že jsem aktuálně jediný pracovník na tomto projektu, tak takový to můj výpadek může mít značné následky.

Nepomáhá tomu ani skutečnost, že mým hlavním relaxačním nástrojem je strongman sport, a zde je riziko zranění jisté zvýšené. Samočřejmě toto je hlavní nebezpečí pokud bude vývoj pouze na jediné osobě.

- Vyšší pravděpodobnost
- Velký dopad

Tabulka 4.8: Matice pravděpodobnosti a dopadu.[11]

Pravděpodobnost	Malý dopad	Menší dopad	Vyšší dopad	Velký dopad
Vysoká			R3	
Vyšší				R1,R4
Menší				
Malá			R2	

Vzhledem k nízké závažnosti rizika R2 ho lze akceptovat, ale kvůli vyššímu možnému dopadu na projekt a možnosti nárůstu pravděpodobnosti v průběhu projektu ho budeme nadále sledovat a připravíme si případný plán pro jeho řízení. U zbylých rizik je již jejich závažnost vysoká a budeme jim tedy věnovat zvýšenou pozornost.

#### 4.7.2 Ošetření rizik

U každého rizika je třeba evidovat jeho vlastníka, který je odpovědný za jeho sledování a hodnocení. Dále je třeba mít plán, jak v průběhu projektu snížit pravděpodobnost, že nastane, nebo snížit jeho dopad. A v poslední řadě je třeba mít také plán pro to, co dělat, pokud dané riziko nastane.

##### 4.7.2.1 Nepřijetí aplikace - R1

Samozřejmě toto riziko se týká až finální části vývoje, ale je třeba ho uvažovat, protože to, jestli je aplikace chtěná a přijímaná, je nejdůležitější faktor pro to, jestli má vůbec smysl ji vyvíjet.

**Vlastník:** Jasenovský Jaroslav

**Mitigace:** Aplikace je vyvíjena dle požadavků a pro potřeby vedoucího mé práce a s tím i celé univerzity. Při vývoji je kladen důraz na jednoduchý a intuitivní návrh uživatelského rozhraní a hlavně snaha o usnadnění práce jejich uživatelů.

**Záložní plán:** Aplikace je založena jako modelář a lze ji snadno přidáním jen malé logické části upravit pro podporu dalších typů modelů, a tím si rozšířit potencionální trh.

##### 4.7.2.2 Zkušenosti - R3

Bohužel nejsem žádný senior vývojář a nemám za sebou desítky projektu, ale mám chuť se vzdělávat a řešení různých problémů je mým koníčkem.

**Vlastník:** Jasenovský Jaroslav

**Mitigace:** Absolvování pomocných kurzů pro vývojáře. Pro samotný vývoj najmout zkušeného vývojáře, aby se rozdělily síly, a zároveň začátečník mohl získávat zkušenosti od pokročilých.

**Záložní plán:** Při nalezení části, kterou nelze úspěšně zvládnout vlastními silami, bude tato část konzultována se zkušenějším vývojářem.

##### 4.7.2.3 Nemoc či zranění - R4

Riziko zranění je zvýšené, ale většinou se jedná o potencionální zranění, která by mě vyřadila zcela z provozu.

**Vlastník:** Jasenovský Jaroslav

**Mitigace:** Zdravý životní styl, vyvážená strava, doplňování vitamínů a minerálů, kvalitní tréninkový plán, zahřívání a protahování, dostatečná regenerace, případná pomocná pracovní síla.

**Záložní plán:** Snaha o co nejrychlejší rekonvalescenci, dočasné pracovní zastoupení.

##### 4.7.2.4 Syndrom vyhoření - R2

**Vlastník:** Jasenovský Jaroslav

**Mitigace:** Pravidelná pracovní doba, volnočasové aktivity (posilování, lezení,...), střídání pracovní náplně, sledování i malých úspěchů.

## 4.8 Závěrečné hodnocení projektu

Vzhledem k tomu, že zatím neexistuje přímá konkurence na trhu, která by zcela podporovala práci s HDM modely, má tento projekt velkou šanci uspět s tímto novým řešením. Navíc je po produktu tohoto projektu poptávka na naší univerzitě. Jak její nasazení na projektech zaměstnanců, tak použití při výuce nabízí další příležitost ve formě uživatelů, které mohou ono řešení prosadit do dalších společností.

Dalším velkým plusem je využívání HDM modelů, které lze aplikovat na systémy různých velikostí a s tím související obtížností. To, společně s plnou podporou skupinové práce, nabízí jednoduché řešení pro firmy zabývající se vývojem technologií.

#### 4.8. Závěrečné hodnocení projektu

---

Proto lze doporučit další podporu daného projektu a po nasazení zkušební verze doladit další možné požadavky, díky kterých bude tato aplikace silnou konkurencí na trhu.





---

## Závěr

V práci jsem se zabýval analýzou, návrhem webové aplikace pro tvorbu HDM a studií proveditelnosti reálného nasazení pro tuto aplikaci.

Díky analýze a vytvořené studii proveditelnosti se prokázalo usnadnění spolupráce na tvorbě modelů. Dále byla ukázána možnost využití ve školách i firmách zabývajících se vývojem systémů, které lze modelovat, a je třeba u nich znát spolehlivost a z ní vycházející Safety Integrity Level.

Hlavní konkurenční výhodou této aplikace je její nativní zaměření na HDM, které splňují požadavky pro návrh téměř všech spolehlivostních systémů. To také umožnilo automatizaci mnoha procesů, díky čemuž usnadní a urychlí práci jejím uživatelům. Nakonec také velice dobře pracuje s různými typy spolupráce a to jak pro potřeby využití při výuce, tak při samotné práci v týmech na firemních projektech.

Díky kompletnímu návrhu a analýze dané aplikace je možné v budoucnu na základě tohoto dokumentu tuto aplikaci implementovat a nasadit do provozu.



---

## Literatura

- [1] HLAVIČKA, Jan. *Číslíkové systémy odolné proti poruchám*. Praha: ČVUT, 1992. ISBN 80-01-00852-5, s. 13–85.
- [2] DANHEL, Martin, Hana KUBATOVA a Radek DOBIA. *Predictive Analysis of Mission Critical Systems Dependability*., vydáno 2013, DOI: 10.1109/DSD.2013.66. ISBN 978-1-4799-2978-8. Dostupné z: <http://ieeexplore.ieee.org/document/6628327/>. (překlad autora). s. 45–52.
- [3] *Moodle-vyuka: BI-TUR* [online]. [cit. 2020-03-08]. Dostupné z: <https://moodle-vyuka.cvut.cz/course/view.php?id=2670>
- [4] *Začínáme s IT: Frontend vs. Backend* [online]. 2014 [cit. 2020-04-19]. Dostupné z: <https://www.czechitas.cz/cs/blog/zaciname-s-it/frontend-vs-backend>
- [5] *Courses: BI-DBS* [online]. [cit. 2020-05-10]. Dostupné z: <https://courses.fit.cvut.cz/BI-DBS/>
- [6] K čemu slouží studie proveditelnosti [online]. [cit. 2020-06-28]. Dostupné z: <https://www.moderniobec.cz/k-cemu-slouzi-studie-proveditelnosti/>
- [7] KEDROŇ, Milan. *Studie proveditelnosti* [online]. Brno, 2010 [cit. 2020-04-27]. Dostupné z: <<https://is.muni.cz/th/k08d4/>>. Diplomová práce. Masarykova univerzita, Ekonomicko-správní fakulta. Vedoucí práce Petr Pirožek.
- [8] WONG, *Before and Beyond Systems: An Empirical Modelling Approach*, kap. 8. PhD thesis. Department of Computer Science, University of Warwick, January 2003. (překlad autora). kap. 8.
- [9] *Creately: State Machine Diagram Tool* [online]. [cit. 2020-05-11]. Dostupné z: <https://creately.com/lp/state-machine-diagram-tool/>. (překlad autora).

## LITERATURA

---

- [10] SAHNER, Robin A. a Kishor S. TRIVEDI. *Reliability Modeling Using SHARPE*. *IEEE Transactions on Reliability* [online]. 1987, R-36(2), 186-193 [cit. 2020-07-29]. DOI: 10.1109/TR.1987.5222336. ISSN 0018-9529. Dostupné z: <http://ieeexplore.ieee.org/document/5222336/>. (překlad autora)
- [11] *Řízení rizik projektu* [online]. [cit. 2020-06-21]. Dostupné z: <https://www.pmconsulting.cz/pm-wiki/rizeni-rizik-projektu/>

## Seznam použitých zkratk

**BORM** Business Objects Relation Modelling

**BPMN** Business Process Model and Notation

**HDM** Hierarchical Dependability Models

**SIL** Safety Integrity Level

### A.1 Slovník pojmů

**Provozní ukazatele** ukazatele produktivity, rychlosti, spotřeby elektrické energie, paliva, apod. [1]

**Spolehlivost** je komplexní vlastnost, která může zahrnovat například poruchovost, životnost, udržitelnost a skladovatelnost, buď jednotlivě, nebo v kombinaci. [1]

**Technické podmínky** je souhrn specifikací technických vlastností, předepsaných pro požadovanou funkci objektu, dále způsoby jeho provozu, skladování, přepravy, údržby a opravy. [1]



---

## Obsah přiloženého CD

readme.txt .....	stručný popis obsahu CD
src	
├ pict .....	obrázky v plné velikosti
├ thesis .....	zdrojová forma práce ve formátu $\text{\LaTeX}$
text .....	text práce
├ thesis.pdf .....	text práce ve formátu PDF