



Online rezervační systém pro lékaře a pacienty na platformě Unicorn Universe

Bakalářská práce

Studijní program: B6209 – Systémové inženýrství a informatika

Studijní obor: 6209R021 – Manažerská informatika

Autor práce: **Daniel Šrám**

Vedoucí práce: doc. Ing. Klára Antlová, Ph.D.



TECHNICKÁ UNIVERZITA V LIBERCI

Ekonomická fakulta

Akademický rok: 2015/2016

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Daniel Šrám**
Osobní číslo: **E13000049**
Studijní program: **B6209 Systémové inženýrství a informatika**
Studijní obor: **Manažerská informatika**
Název tématu: **Online rezervační systém pro lékaře a pacienty na platformě Unicorn Universe**
Zadávací katedra: **Katedra informatiky**

Z á s a d y p r o v y p r a c o v á n í :

1. Platforma Unicorn Universe - koncepce a funkcionality
2. Nástroje Unicorn - metodika a modelovací jazyk
3. Online rezervační systémy
4. Návrh a realizace systému pro rezervaci pacientů
4. Zhodnocení řešení

Rozsah grafických prací:

Rozsah pracovní zprávy: **30 normostran**

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

KOVÁŘ, Vladimír a aj. Unicorn ES Powered Company - Management. Praha: Unicorn College, 2009. ISBN 978-80-87349-01-4.

GÁLA, Libor, Jan POUR a Zuzana ŠEDIVÁ. Podniková informatika. 2., přeprac. a aktualiz. vyd. Praha: Grada, 2009. ISBN 9788024726151.

FROST, Raymond & PIKE, Jacqueline. Business information systems: design an app for that. Version 1.0. Washington, DC: Flat World Knowledge, 2013. ISBN 978-1-4533115-7-8.

Elektronická databáze článků ProQuest (knihovna.tul.cz).

Vedoucí bakalářské práce:

doc. Ing. Klára Antlová, Ph.D.

Katedra informatiky

Konzultant bakalářské práce:

Ing. Lucie Melounová

VIGOUR a. s. Praha

Datum zadání bakalářské práce: **31. října 2015**

Termín odevzdání bakalářské práce: **31. května 2017**



doc. Ing. Miroslav Žižka, Ph.D.
děkan



doc. Ing. Jan Skrbek, Dr.
vedoucí katedry

V Liberci dne 31. října 2015

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum:

Podpis:

PODĚKOVÁNÍ

Touto cestou bych rád poděkoval vedoucí mé bakalářské práce doc. Ing. Kláře Antlové, Ph.D. za vedení, cenné rady a inspiraci v podobě doporučených zdrojů. Dále bych rád poděkoval Bc. Janu Rudolfovi za odborné rady a ochotu, se kterou mne po celou dobu praxe učil novým věcem.

ANOTACE

Cílem této bakalářské práce je návrh a vývoj Online rezervačního systému pro lékaře a pacienty na platformě Unicorn Universe. Klíčovou myšlenkou aplikace je provést návrh a realizaci tak, aby co nejvíce usnadnil práci lékařům a čas pacientům. Slouží k objednávání pacientů do systému s možností, aby byl pacient schopen objednat se sám z pohodlí svého domova. Dále aplikace slouží jako základní kámen nového ambulantního informačního systému pro lékaře.

V teoretické části je představena digitální stavebnice informačních systémů Unicorn Universe, k jejímuž plnému pochopení je třeba porozumět používané metodice společnosti. Proto v této části budou vysvětleny pojmy jako Unicorn Approach, Unicorn Process, Unicorn Universe Business Modelling Language a další.

Praktická část se zabývá návrhem rezervačního systému stojícího na platformě Unicorn Universe, jeho realizací a zhodnocením přínosu analyzovaného řešení.

KLÍČOVÁ SLOVA

Unicorn Universe, Rezervační systém, Plus4U

ANNOTATION

The aim of this thesis is design and development of the Online Reservation System for Patients and Doctors using the Unicorn Universe Platform. The purpose of the application itself is to allow its users to be able to make medical appointments from the comfort of their own homes. Furthermore, the application will serve as a building block of the new medical information system.

The theoretical part introduces the digital construction set called Unicorn Universe. For the complete understanding of mentioned platform, several issues must be mentioned such as company methodology. For that reason, terms like Unicorn Approach, Unicorn Process, Unicorn Universe Business Modelling Language and more are explained there.

The practical part of the thesis deals with the design, development and contribution evaluation of the Online Reservation System itself.

KEY WORDS

Unicorn Universe, Reservation system, Plus4U

Obsah

Seznam zkratk	9
Seznam obrázků	10
ÚVOD	11
Literární rešerše	12
1. PLATFORMA UNICORN UNIVERSE	13
1.1 Unicorn Approach	13
1.2 Digitální stavebnice informačních systémů Unicorn Universe	15
1.3 Služba Plus4U	18
2. NÁSTROJE UNICORN	19
2.1 Unicorn Universe Process	19
2.2 Unicorn Universe Operating System	20
2.3 Unicorn Universe Applications	21
2.4 Unicorn Universe Business Modeling Language	22
3. METODOLOGIE VÝVOJE	24
3.1 Agilní metodiky vývoje	24
3.2 Porovnání metodiky Scrum a Unicorn Approach	27
4. ONLINE REZERVAČNÍ SYSTÉM - NÁVRH	29
4.1 Zhodnocení současného stavu	30
4.2 Klíčová myšlenka	30
4.3 Produktový pohled	32
4.4 Analýza případů užití	33
4.5 Návrh rezervačního widgetu	36
5. ONLINE REZERVAČNÍ SYSTÉM - REALIZACE	38
5.1 Naplánovat prohlídku	38

5.2 Rezervační widget.....	40
5.3 Zhodnocení přínosu	41
ZÁVĚR	42
Seznam použité literatury	43
Seznam příloh.....	45

SEZNAM ZKRATEK

IS	Informační systém
UU	Unicorn Universe
UA	Unicorn Approach
OJ	Organizační jednotka
API	Application Programming Interface
BB kód	Bulletin Board Code
uuOS	Unicorn Universe Operating System
uuBML	Unicorn Universe Business Modeling Language
VUC	Visual Use Case
PO	Product Owner
REST	Representational State Transfer
PSIs	Potentially Shippable Increments

SEZNAM OBRÁZKŮ

Obrázek 1: Artefakt

Obrázek 2: Služba Plus4U

Obrázek 3: 12 procesů metodiky uuProcess

Obrázek 4: Výchozí předpoklady pro vývoj aplikace uuApp

Obrázek 5: Ukázka ikon jazyka uuBML

Obrázek 6: Role Product Ownera

Obrázek 7: Klíčová myšlenka rezervačního systému

Obrázek 8: Znárodnění vazeb mezi objekty systému

Obrázek 9: Ukázka popisu případu užití – Vytvořit sestru

Obrázek 10: Návrh rezervačního widgetu

Obrázek 11: Realizace případu užití – Naplánovat prohlídku

Obrázek 12: Třívrstvá architektura rezervačního widgetu

ÚVOD

Informační systém je v dnešní době nezastupitelnou součástí podniku a na proces jeho vývoje existuje mnoho pohledů. Rychle se měnící situace na trhu si navíc říká o rychlá a moderní řešení jeho implementace. Při úvahách o informačním systému však nesmíme opomenout neodmyslitelnou součást jakékoli aplikace, jíž jsou uživatelé. Specifickým příkladem takového IS je pak rezervační systém určený pro lékaře a pacienty. Zmíněný produkt vyžaduje jednoduchost v užívání na straně pacienta. Zároveň však musí být dostatečně komplexní, aby zvládl efektivně pracovat s veškerými daty, která lékaři pro výkon své profese potřebují. Zdárná implementace takového informačního systému je jeden ze způsobů, jak uspokojit stále rostoucí poptávku po zdravotnické péči.

Tato bakalářská práce se zabývá návrhem a realizací online rezervačního systému pro lékaře a pacienty na platformě Unicorn Universe. Aplikace popsána v této práci je vyvíjena zákazníkovi na míru a bude sloužit primárně k rezervacím. Sekundárním účelem této aplikace je sloužit jako základní kámen pro budoucí ambulantní informační systém. Její struktura tedy počítá s tím, že na ni budou navazovat další moduly v podobě Unicorn Universe aplikací. Modul zmíněného systému bude schopen obstarat vytvoření ambulantního zařízení, ordinace, zdravotnického personálu a pacientů v platformě UU a především zajistit vytváření objednávek.

V teoretické části bakalářské práce jsou uvedeny a vysvětleny principy metodiky Unicorn Approach, jenž slouží k efektivnímu řízení všech podnikových procesů. Cílem následujících kapitol je seznámit čtenáře se základními kameny platformy Unicorn Universe, která uvedenou metodiku podporuje.

V praktické části je zhodnocena současná situace na trhu ambulantních informačních systémů. Práce dále pokračuje představením idey aplikace, po které již následuje její samotný návrh. V další části jsou uvedeny vybrané problémy implementace a je tak popsána realizace projektu.

LITERÁRNÍ REŠERŠE

Kapitola Literární rešerše uvádí přehled a stručný popis děl, která se zabývají podobnou problematikou jako tato bakalářská práce. Jedná se tedy především o články a literaturu zaměřenou na rozbor, návrh či implementaci informačních systémů. Větší pozornost je kladena na díla zaměřená na zdravotnické informační systémy.

Článek *Adopting electronic medical records in primary care: Lessons learned from health information systems implementation experience in seven countries* od autorů D. A. Ludwicka a J. Doucetteho popisuje vlivy působící na výsledek implementace zdravotnických informačních systémů. Zmíněné dílo dále uvádí způsoby, jakými lze zmírnit riskantní faktory procesu implementace IS, mezi nimiž lze nalézt například standardizaci lékařské terminologie. V neposlední řadě článek předkládá nejčastější argumenty lékařů, kteří implementaci informačního systému odmítají.

Obecný přehled informačních systémů pro lékařské účely je uveden v článku *Health Information Systems*, jehož autory jsou Joseph Sirintrapun, M.D. a David Artz, M.D. V této práci jsou zdravotnické IS chápány jako komplexní systémy, které zachycují, shromažďují, třídí a prezentují rozsáhlé množství dat souvisejících s klinickou péčí. Takto definované systémy autoři dále dělí na obecné systémy, finanční, úsekové (zaměřené na určité oddělení) a systémy pro elektronickou evidenci lékařských záznamů.

Dalším přínosným čtením pro účely této bakalářské práce je disertační práce Ing. Vladimíra Kováře nesoucí název *Metodika Unicorn Enterprise System Powered Company*. V této práci je podrobně rozebrána a na příkladech vysvětlená metodika Unicorn Universe Process. Na uvedených principech je založena platforma Unicorn Universe, a tím pádem i aplikace online rezervačního systému pro lékaře a pacienty, jež je předmětem této práce.

1. PLATFORMA UNICORN UNIVERSE

1.1 Unicorn Approach

Společnosti skupiny Unicorn, které se zabývají především vývojem informačních systémů a řešením na míru zákazníkům, vyčnívají svým specifickým přístupem k řízení svých zaměstnanců a dceřiných společností. Tento přístup pevně stojí na několika základních myšlenkách, které jsou léty praxe prokázány jako funkční a efektivní a které společnost Unicorn v řízení sama aplikuje a které pod označením Unicorn Approach dále předává jako produkt v podobě unikátního know-how.

Unicorn Approach je tedy soubor doporučených postupů, myšlenek, principů a zásad, které jsou aplikovány při tvorbě informačních systémů. Sám autor tohoto přístupu a majitel společnosti Unicorn Ing. Vladimír Kovář formuloval následující hypotézu:

„Domnívám se (vlastně jsem o tom hluboce přesvědčen), že je možné v celé řadě případů podrobnějším řízením, nežli je v současnosti obvyklé, získat podniku konkurenční výhodu. Zároveň jsem však přišel na to, že bez potřebné metodiky a její přímé podpory informačním systémem podrobněji, nežli je obvyklé, řídit nelze.“ [1]

Soupis pravidel Unicorn Approach lze rozdělit do deseti základních oblastí. Nutné je ještě zmínit, že souhrn těchto deseti klíčových myšlenek a principů se projevuje ve všech činnostech firmy a tvoří tak základní stavební pilíře dalších pravidel Unicorn Approach. Pořadí myšlenek vychází z jejich důležitosti a zároveň toho platí, že jednotlivé myšlenky spolu navzájem souvisí.

První a nejdůležitější myšlenkou je **systemový přístup**. Přístup Unicorn Approach totiž věří, že systém zachovává pořádek ve všech činnostech probíhajících ve firmě, a pokud každý věnuje pět procent svého času na udržování pořádku, celková výkonnost se o třicet procent zvedne. Systémem zde ovšem rozumíme souhrn prvků a jejich vazeb, které jsou organizovány za účelem splnění určitého cíle (v případě podnikání bude tento cíl především zisk). Systemový přístup spočívá v tom, že k problému, který řešíme, vytvoříme systém a snažíme se jeho chování optimalizovat tak, abychom našli řešení, které hledáme. Podnik chápeme jako systém vzájemně spolupracujících

organizačních jednotek, jehož cílem je dosáhnout stanovené strategie. Jednotky spolu komunikují, sdílí informace a zvláště si navzájem poskytují své produkty a služby.

Druhou myšlenkou je **delegace kompetencí**, jejíž praktické provedení je podrobněji vysvětleno v kapitole Unicorn Universe. Unicorn Approach zaměstnance motivuje k předávání kompetencí a zodpovědnosti jiným, třeba i mladým lidem. Delegování kompetence není jen o předávání pravomocí a odpovědnosti na druhé. Ten, kdo deleguje, musí být přesvědčen, že podřízený je schopen úkoly splnit, a v žádném případě se delegováním nezbavuje své vlastní odpovědnosti. Umím si představit scénář, ve kterém čtenář bude delegaci kompetencí chápat negativně. Ve společnosti Unicorn to tak ale není, z vlastní zkušenosti mohu říci, že delegací nějaké aktivity na mě, jako na méně zkušeného kolegu, cítím důvěru nadřízeného v mé schopnosti a celé to vnímám velmi motivačně.

Třetí pravidlo by se dalo formulovat jako **orientace na zákazníka a zaměstnance**. Zaměstnancům je potřeba věnovat stejnou pozornost jako zákazníkům, protože kvalitní služby jsou na kvalitních a spokojených zaměstnancích postavené. Orientace na zákazníka a zaměstnance jsou vzájemně spojeny do jednoho bodu - zdůrazňuje lidský přístup - software dělají lidi pro lidi. V orientaci na zákazníky platí, že je třeba dělat takové věci, které mají smysl. Vždy je potřeba pochopit podstatu problému zákazníka a neomezovat se na technologická řešení. Je samozřejmě žádoucí, aby to, co má smysl, bylo totožné s tím, co zákazník chce. Je nutné vzájemně sladit zákaznický pohled na problém a pohled, kterým se na problém dívá Unicorn. Jak zákazník, tak Unicorn nějakým způsobem dekomponují zákaznickův problém a ve finále je nutné, aby byly obě dekompozice problému (tj. způsoby řešení) konzistentní. V praxi je tato myšlenka řešena pravidelnými schůzkami (kontrolními dny) na nichž se představuje vendorovi (zástupci zákazníka) doposud vyvinuté řešení a diskutuje se další postup.

Dalšími myšlenkou je **cílené nasazení**, jenž vlastně říká jen to, že v každé fázi vývoje produktu by si měli být všichni zúčastnění vědomi toho, co je cílem, a dělat maximum pro to, aby se k němu co nejdříve dostali. Dále Unicorn Approach obsahuje princip **standardizace procesů**, který svým uživatelům doporučuje standardizovat veškeré předem známé procesy. Pokud si firma vytvoří standardy, umožní jí to dělat opakované věci rychleji, měřit je, srovnávat a zlepšovat. Firmy společnosti Unicorn mají všechny známé scénáře řešení problémů standardizované. Ve vývoji aplikací, jehož jsem součástí je toto pravidlo hojně využíváno. Všichni zaměstnanci řeší stejné

problémy stejně, nesnaží se vymýšlet to, co již bylo vymyšleno, a díky tomu pak zbývá více času na řešení nových, zatím neznámých problémů.

V neposlední řadě myšlenek Unicorn Approach stojí **průběžné hodnocení**. To se týká nejen celého projektu v podobě již zmíněných kontrolních dní ale také každého jednotlivého zaměstnance, který se každé tři měsíce sám hodnotí a zároveň je hodnocen svým přímým nadřízeným. Je totiž třeba neustále ověřovat, že to, čím se firma nebo projekt aktuálně zabývá, má smysl. Průběh činnosti je porovnáván s plánem a zejména je třeba řídit očekávání klienta - připravovat jej na to, co dostane, a ověřovat, jestli to, co dostane, je to, co opravdu chce a očekává. Nejlepší plán je takový, který je měřitelný.

Dalším principem je **sdílení informací a znalostí**. V dnešním rychle se rozvíjícím světě je toto pravidlo velmi důležité. Informace a znalosti je třeba ve firmě sdílet. Také je potřeba oddělit znalosti od lidí a uložit je na centralizovaném místě, kde budou přístupné třeba i po odchodu klíčových zaměstnanců. Tento princip úzce souvisí se systémem pracovních rolí, který je podrobněji rozebrán a na příkladu vysvětlen v kapitole Unicorn Universe (viz str. 16).

Předposledními myšlenkami jsou **paměť a poučení** (učení se z chyb a neopakování stejných chyb) a **soustavné zlepšování**. V praxi jsou mimo jiné aplikovány pomocí registru projektů, kde je ke každému projektu uveden popis procesů s vyhodnocením, ze kterých se může čtenář poučit.

Posledním principem přístupu Unicorn Approach je **stabilita a bezpečnost**. Tato myšlenka ve stručnosti říká že, i přestože je vyšším principem snaha stále růst, rozvíjet se a vydělávat více, je třeba zároveň neplýtvat zdroji a předcházet možným rizikům. Růst firmy musí vycházet z pevných a stabilních základů a musí stát na přiměřeně robustní a bezpečné struktuře.

1.2 Digitální stavebnice informačních systémů Unicorn Universe

Unicorn Universe je digitální stavebnice informačních systémů, jenž vychází z myšlenek Unicorn Approach využívající metodiku Unicorn Universe Process a její řešení jsou poskytována pomocí služby Plus4U. V několika následujících kapitolách je vysvětleno, jak tato platforma funguje a jakými objekty je tvořena.

1.2.1 Teritorium

Pojem teritorium lze chápat z několika úhlů pohledu. Fyzicky je teritoriem sektor disku, tedy část paměti vyhrazená právě pro dané teritorium. V systému je potom teritorium prostor určený buď soukromému uživateli (v tomto případě jej nazýváme My Territory), nebo prostor určený živnostníkovi, firmě či společnosti. Takový subjekt může toto místo využívat, jak chce, a v případě nedostatku kapacity pro nová data je nucena zaplatit si za rozšíření svého teritoria. V tomto případě pak mluvíme o Business Territory, které je dále členěno pomocí organizačních jednotek.

1.2.2 Organizační jednotka

Jednotlivé organizační jednotky v teritoriu odrážejí reálnou organizační strukturu podniku. Tyto jednotky (např. Ekonomické oddělení, Personální oddělení, atd.) mohou být pro přehlednost dále členěny pomocí složek a mohou obsahovat podřízené jednotky. Převážně ale fungují jako ucelené balíčky informací a vztahují se k nim specifická práva (na čtení, zápis či spouštění funkcí) a role.

1.2.3 Role a skupiny

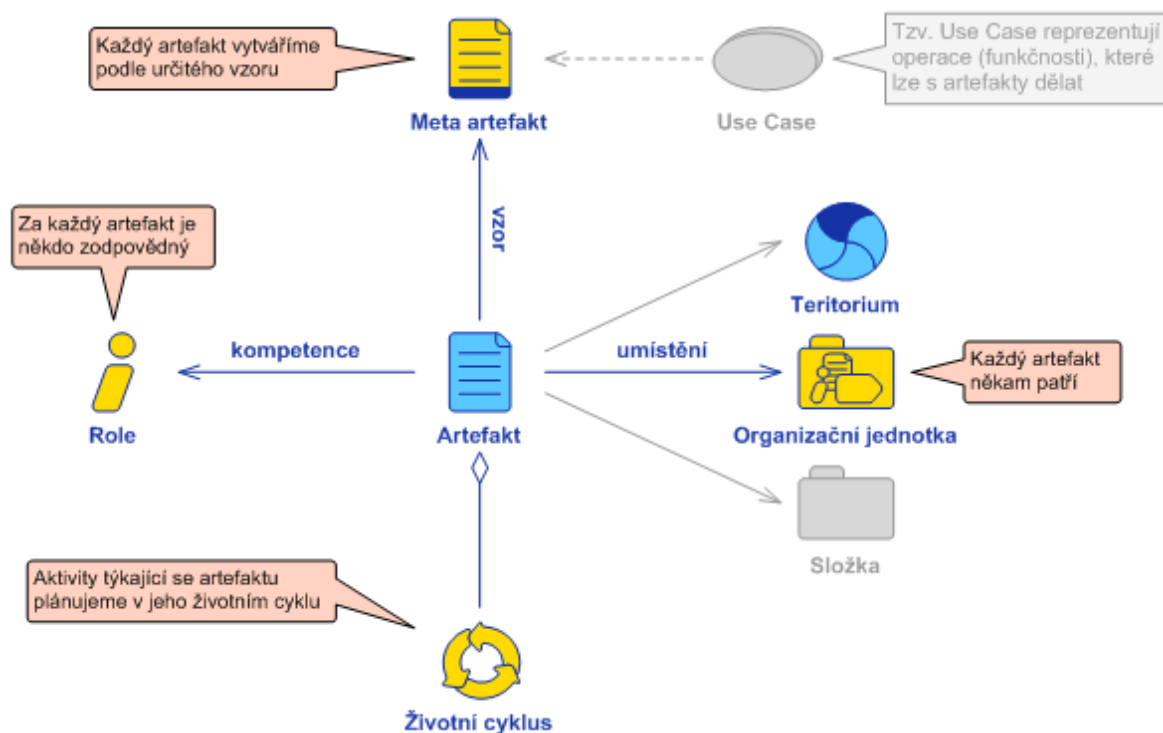
Na platformě Unicorn Universe existuje několik druhů rolí a skupin a jistě by se o nich dalo napsat mnoho, v této práci je ovšem uvedeno jen nutné minimum pro pochopení jejich fungování. Role jsou rozděleny na dva základní typy – přístupové a pracovní. Každý uživatel, jenž využívá služby Plus4U, je zároveň majitelem své personální role, řadící se pod přístupovou roli. Právě pomocí této osobní přístupové role je dále uživatel obsazován do přístupových rolí v jiných teritoriích za účelem získání přístupu.

Druhým typem rolí jsou role pracovní, které, jak již název napovídá, slouží k vykonávání určité pracovní role. Společně s obdržetím pracovní role uživatel obvykle získá určité kompetence a zodpovědnost stejně jako v reálném světě. Skupiny slouží k hromadnému přidělování práv jednotlivým rolím, jenž jsou součástí dané skupiny.

1.2.4 Artefakt a Meta artefakt

Artefakt je základním nositelem informace. Obecně platí, že vše v systému Unicorn Universe je buď artefakt, nebo jeho agregovaný objekt. Meta artefakt je předchůdcem artefaktu a určuje jeho základní vlastnosti (vzorový životní cyklus, možné stavy, aktivity nad artefaktem vytvářené a vzhled). I meta artefakt má svého předchůdce a vše, co bylo výše zmíněno (organizační jednotky, složky, role a skupiny) jsou právě artefaktem. Artefakt totiž reprezentuje objekty reálného světa a to jak fyzické jako například budovy či lidi, tak i objekty nehmataelné (know-how, cíle, obchodní příležitosti a jiné). Za každý artefakt je odpovědná nějaká role.

Artefakt se skládá ze dvou částí – obsahu a životního cyklu. Obsah artefaktu znázorňuje jeho informace. Životní cyklus slouží k řízení artefaktu a jeho vzor je vždy definován na meta artefaktu. Pomocí životního cyklu jsou řízeny a nastavovány stavy daného artefaktu.

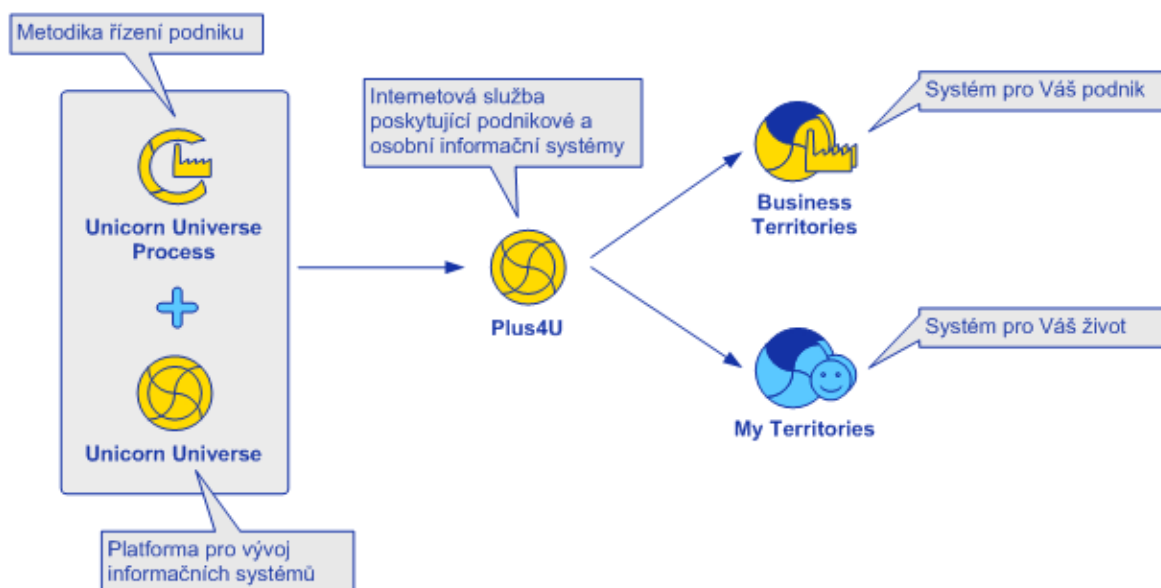


Obrázek 1: Artefakt

Zdroj: Interní dokumentace společnosti Unicorn

1.3 Služba Plus4U

Plus4U je internetová služba fungující na platformě Unicorn Universe, pomocí níž jsou poskytována softwarová řešení na míru. Výhodou internetové služby jest fakt, že informační systém je jeho uživatelům dostupný neustále a odkudkoli, tedy za předpokladu, že mají přístup k internetu. Registrace je zdarma a soukromý uživatel v ní najde základní balíček klasických plánovacích aplikací a úložného prostoru, korporátní zákazníci, jenž chtějí tímto informačním systémem obohatit své podnikání si již za tuto službu musí zaplatit. V současné době tuto službu využívá 54 000 uživatelů. [2]



Obrázek 2: Služba Plus4U

Zdroj: Unicorn Universe [online]. 2008, 2012 [cit. 2015-12-28]. Dostupné z: <https://unicornuniverse.eu/>

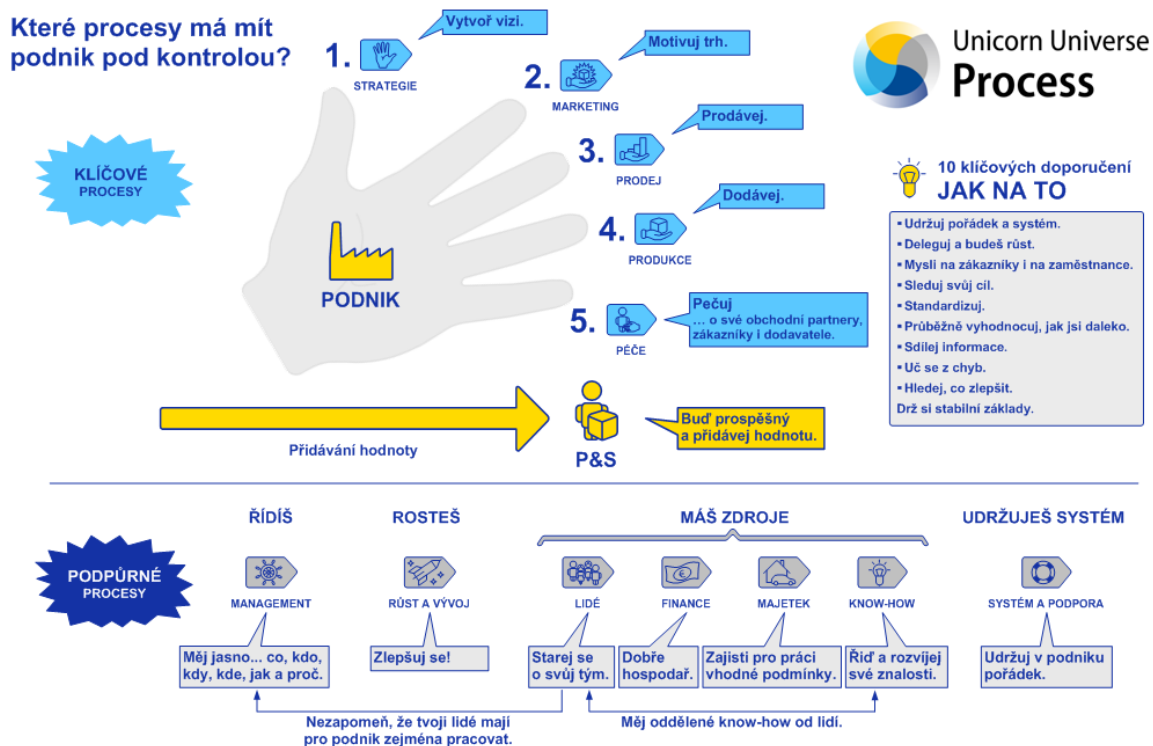
2. NÁSTROJE UNICORN

2.1 Unicorn Universe Process

Metodika Unicorn Universe Process (dále jen uuP) je obecný pracovní postup nabízející podrobný návod, jak řídit veškeré podnikové procesy s použitím korporátního informačního systému vybudovaného nad platformou Unicorn Universe. Základem této metodiky jsou myšlenky, které společnosti Unicorn ve svém řízení užívají. Tato metodika je dále pod označením Unicorn Approach poskytována jako produkt, tedy unikátní know-how ověřené dlouholetou praxí.

Díky univerzálnosti a obecnosti uuP lze tento přístup využít pro jakýkoli obor podnikání. Kromě obecných doporučených postupů ovšem uuP zahrnuje i návody pro konkrétní procesní oblasti (například oblast výroby). Další předností uuP je její rychlá implementace. Cílem této metodiky je standardizace procesů, která umožňuje dělat věci rychleji.

Metodika uuP rozlišuje dvanáct podnikových procesů, z nichž pět označuje jako klíčové a zbylých sedm jako podpůrné, toto rozdělení znázorňuje obrázek č. 3.



Obrázek 3: 12 procesů metodiky uuProcess

Zdroj: dokumentace služby Plus4U

2.2 Unicorn Universe Operating System

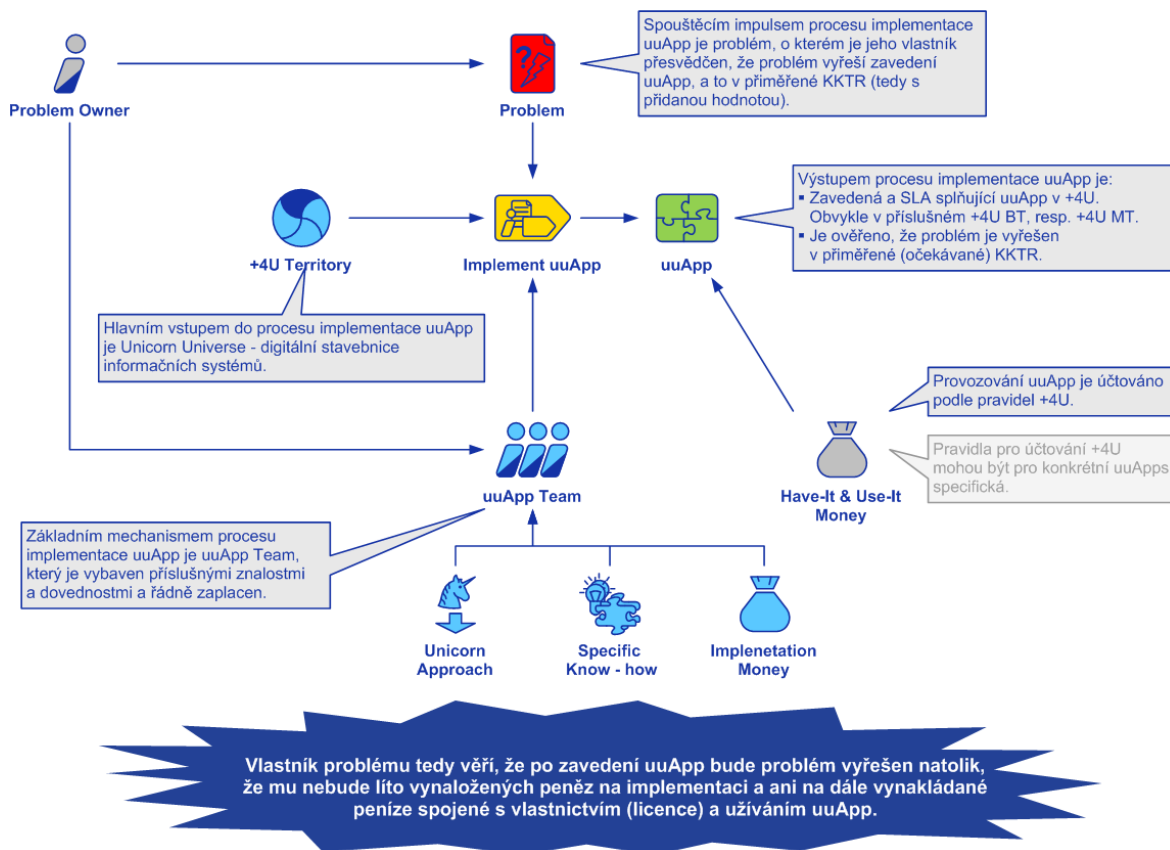
Unicorn Universe Operating Systém (dále jen uuOS) je univerzální platforma pro vývoj a provoz komplexních informačních systémů. Umožňuje paralelní provoz informačních systémů různých uživatelů v cloudů. Samozřejmostí je implementace metodiky uuP, tím pádem usnadňuje komunikaci, řízení podnikových procesů a podporuje správu a sdílení informací. Platforma uuOS je základním předpokladem pro vývoj aplikací uuApps.

2.3 Unicorn Universe Applications

Unicorn Universe Applications (dále jen uuApps) jsou součástí platformy Unicorn Universe, které mohou svou velikostí zastat jak malé, užitečné aplikace (například na třídění požadavků), tak rozsáhlé informační systémy (například Ambulantní informační systém).

Aplikace uuApps využívá softwarové architektury MVC (Model View Controller), která rozděluje datový model, uživatelské rozhraní a řídicí logiku do tří poměrně nezávislých komponent. Uživatelské rozhraní je řešeno pomocí tzv. vizuálních případů užití, které vytvoří designér z již předpřipravených komponent. Řídicí logika je konstruována pomocí programovacího jazyka Ruby a datový model je řešen technologií MongoDB. MongoDB je open-source databáze, ve které jsou data reprezentována jako dvojice klíč a hodnota. Nejedná se o klasický model entity a vztahu, který již v dnešní době není dominantním řešením v návrhu datového modelu. [3]

Aplikace uuApps se dělí na několik druhů. Nejzákladnějším typem je Common uuApp, tedy aplikace instalovaná jako standardní součást BT (Business Territory). Standard uuApp je aplikace instalovaná z uuApp Store. Custom uuApp je aplikace instalovaná zákazníkovi na míru. Single-Unit uuApp je aplikací určenou pro provoz v jednom BT s jednou konfigurací a Multi-Unit uuApp je aplikace schopná provozu ve více instancích s vlastními konfiguracemi v rámci jednoho teritoria.



Obrázek 4: Výchozí předpoklady pro vývoj aplikace uuApp

Zdroj: školící materiály Top Gun Academy

2.4 Unicorn Universe Business Modeling Language

Unicorn Universe Business Modeling Language (dále jen uuBML) je modelovací jazyk, pomocí něhož jsou navrhovány a dokumentovány aplikace. Lze jím ale zakreslit libovolnou problematiku či algoritmus z běžného života.

uuBML je inspirováno vizualizačním jazykem UML (Unified Modeling Language), ale jednotlivé objekty jsou v něm reprezentovány speciálními ikonami. Díky tomu je uuBML velmi intuitivní a přehledné, důležitost jednotlivých objektů je navíc barevně rozlišena (zelená > žlutá > modrá > tmavě šedá > světle šedá).



Obrázek 5: Ukázka ikon jazyka uuBML

Zdroj: dokumentace služby Plus4U

3. METODOLOGIE VÝVOJE

Společnost Unicorn je poměrně výjimečná ve svém stylu řízení a podobné přístupy v jiných firmách se hledají obtížně. Jisté podobnosti lze ovšem nalézt v hojně rozšířených metodologiích, které jsou využívány projektovými týmy po celém světě. V této kapitole jsou uvedeny v současnosti nejvyužívanější metodologie vývoje softwaru a následně je uvedeno jejich srovnání vůči systému vývoje používaného společností Unicorn.

Nebude jistě žádným překvapením, že se tato kapitola zabývá pouze agilními metodologiemi vývoje. Výzva, před níž v dnešní době stojí každá softwarová společnost, spočívá ve schopnosti být konkurenceschopný za účelem udržení si stávajícího místa na trhu, nebo ještě lépe, jeho rozšíření. Rigorózní přístupy k vývoji softwaru jsou neflexibilní a selhávají v rychlé reakci na rychle se měnícím trhu. Naproti tomu agilní přístupy k vývoji softwaru nabízejí celou řadu postupů, které napomáhají obratné reakci na stále se měnící podmínky a splňují tak požadavky moderního vývoje produktu.

Případová studie provedená univerzitou Hellenic Open University v Řecku potvrzuje, že agilní přístupy k vývoji softwaru poskytují lepší výsledky než jejich tradiční předchůdci, a to dokonce i na rozsáhlých, distribuovaných projektech. Zmíněné lepší výsledky lze pozorovat na hodnocení produktu zákazníkem, jelikož agilní přístupy umožňují snadnější změnu požadavků v průběhu vývoje. Zároveň se vlivem zvýšené komunikace a vzájemné spolupráce v rámci vývojového týmu zvyšuje i spokojenost zaměstnanců. [4]

3.1 Agilní metodiky vývoje

Agilní metodiky vývoje si dávají za cíl rychlý vývoj a následnou reakci na změny. Jak již bylo zmíněno, trend spěje k agilnímu vývoji, poněvadž vznikl jako reakce na nedostatky tradičních přístupů, a podporuje tak efektivitu vývojového procesu. Následuje výňatek z manifestu agilního vývoje, který obsahuje myšlenky sloužící jako stavební kameny všech agilních přístupů.

„Odhalili jsme lepší způsob vývoje software, sami jej používáme a chceme pomoci i ostatním, aby jej používali. Z tohoto pohledu dáváme přednost:

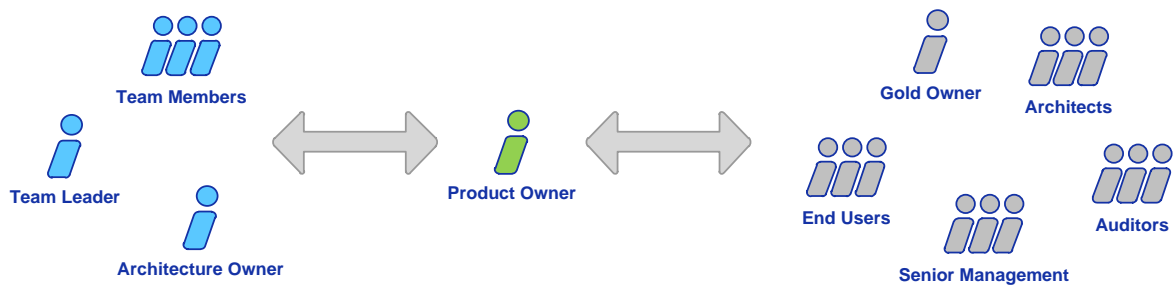
- *individualitám a komunikaci před procesy a nástroji,*
- *provozuschopnému software před obsažnou dokumentací,*
- *spolupráci se zákazníkem před sjednáváním kontraktu,*
- *reakci na změnu před plněním plánu.“ [5]*

3.1.1 Scrum

Metodika Scrum stojí na několika základních principech agilního vývoje, a dává si tak za cíl dodat produkt v krátkém vývojovém cyklu, který podporuje rychlou zpětnou vazbu, kontinuální zlepšování a okamžitou reakci na změnu. Tato metodika poskytuje flexibilní strategii vývoje produktu, při které vývojový tým, jenž obvykle tvoří tři až devět lidí, pracuje na dosažení společného cíle. Dále vyvrací předpoklady rigorózního přístupu a tvrdí, že tým se dokáže organizovat sám díky blízké spolupráci svých členů. [6]

3.1.1.1 Role

Scrum tým se skládá ze tří základních rolí. **Vlastník produktu** (Product owner) reprezentuje po dobu celého vývojového procesu zákazníka a jeho hlavním úkolem je dohlížet na to, aby se finální produkt splňoval co nejpřesněji přání zákazníka. Zároveň má PO v kompetenci stanovení priorit a komunikaci se zákazníkem. PO je vlastně takový obojživelník, jelikož pro vývojový tým představuje zástupce zainteresovaného subjektu v produktu (tzv. stakeholder) a zároveň pro skutečné investory hraje roli představitele vývojového týmu. [7] [8]



Obrázek 6: Role Product Ownera

Zdroj: vlastní

Druhou rolí je **vývojový tým** (Scrum team), jehož primárním posláním je pomocí několika relativně funkčních inkrementů (potentially shippable increments - PSIs) na konci každého sprintu dodat finální produkt [8]. Jednotliví členové takového týmu se zabývají činnostmi, jako jsou analýza, design, vývoj, testování a zpracovávání dokumentace. Poslední definovanou rolí je **Scrum master**, jenž odstraňuje jakékoli překážky, které se mohou na cestě k finálnímu produktu vyskytnout. Dále tato osoba organizuje schůzky a motivuje tým k lepším výkonům.

3.1.1.2 Eventy

Metodika scrum rozlišuje dvě základní události (eventy), které jsou součástí procesů vývoje finálního produktu. Prvním z nich je **sprint**, někdy nazýván prostě jako iterace. Sprint je základní časová vývojová jednotka, je to časově vymezená iterace vývojového cyklu, v rámci kterého má vývojový tým splnit stanovené cíle. Tato iterace obvykle trvá jeden týden až jeden měsíc.

Druhým eventem je **denní scrum** (nebo stand-up). Jedná se o krátkou schůzku (obvykle do 15 minut), která začíná každý den ve stejný čas a na stejném místě. Členové týmu chodí na schůzku připraveni a týmové role scrumu mluví o dosažených výsledcích. Podle dat z případové studie provedené na univerzitě v Oslu jsou jako pozitivní faktory tohoto eventu nejčastěji vnímány sdílení informací a příležitost kolektivního řešení problémů. Jako faktory přispívající k negativnímu vnímání těchto schůzek respondenti nejčastěji uvádějí nutnost dostavit se na schůzku každý den a příliš velkou délku schůzky. [9]

3.2 Porovnání metodiky Scrum a Unicorn Approach

Ze všech agilních přístupů k tvorbě softwaru byla pro účel této bakalářské práce vybrána metodika Scrum právě proto, že v ní lze nalézt mnoho podobností s vývojem řízeným podle zásad Unicorn Approach. Už jen v základních myšlenkách obou přístupů jsou identické zásady jako například **soustavné zlepšování** a **průběžné hodnocení**. Viditelný je i společný přístup k chápání týmu, kde scrum tým definuje jako skupinu lidí jdoucí za stejným cílem. Stejnou myšlenku zastává i Unicorn Approach a definuje ji v zásadě **cílového nasazení**, jejíž vysvětlení lze nalézt v kapitole Platforma Unicorn Universe (viz str. 14). Realizace některých myšlenek je již ovšem odlišná.

3.2.1.1 Role

Základní vývojovou jednotkou UA je pár designér-developer. Designér zastává podobnou roli jako PO s tím rozdílem, že je designér větší součástí vývojového týmu. V kompetenci designéra je zároveň i údržba dokumentace, řešení případných logických problémů návrhu aplikace a testování. Práce developera spočívá v psaní kódu a tvorbě datového modelu.

Role Scrum mastera lze lehce srovnat s rolí Team leadera. Od metodiky scrum se role Team leadera v UA liší snad jen tím, že Team leader řídí současně prakticky všechny projekty, které spadají pod jeho vývojovou jednotku. Například vývojová jednotka Neatcode má v současné době 18 pracovníků a implementuje odhadem deset projektů v týmech rozdělených po dvou až pěti pracovnících. Jeden designér obvykle zvládá být součástí více týmu a Team leader je díky tomu schopen mít pod kontrolou všechny projekty a poskytovat cenné rady.

3.2.1.2 Události

Vývoj softwaru pod zásadami UA je prakticky stejný jako v metodice scrum. Pracuje se v iteracích, které jsou dlouhé jeden týden až jeden měsíc. Po každé iteraci je v rámci kontrolního dne vyvíjený produkt prezentován zástupci zákazníka, který může na aplikaci vznést nové nebo změnové požadavky, které jsou následně po domluvě implementovány. Takovýto prezentovaný produkt představuje funkční modul aplikace (podobně jako PSIs) a v kontrolní den se předává testovacímu týmu.

Zatímco v metodice scrum se vývojový tým schází každý den, UA v rámci průběžného hodnocení předepisuje schůzku jednou týdně. Tato schůzka se nazývá Status Assessment (SA) a účastní se jí všechny vývojové týmy dané vývojové jednotky. Stejně jako při stand-upu se zde probírají problémy, které kdo řeší a sdílí se informace s tím rozdílem, že SA trvá obvykle jednu hodinu a účastní se ho i ředitel vývojové jednotky.

3.2.1.3 Shrnutí

Obě popisované metodiky jsou bezpochybně použitelné v moderním a rychle se měnícím světě a stojí na základech agilního přístupu. Největším rozdílem je pravděpodobně styl řízení. Zatímco metodika scrum podporuje samořízení týmu, Unicorn Approach vychází z předpokladu podrobnějšího řízení podpořeného informačním systémem.

Unicorn Approach ovšem obsahuje jednu zásadu, která se lehce vymyká nejen metodice scrum, ale zároveň celému manifestu agilního vývoje. Touto zásadou je standardizace procesů, podle kterého je nejlepší standardizovat všechny známe procesy ve firmě a připravit se tak na jejich řešení. Toto pravidlo ovšem není implementované pouze do řízení společnosti, ale i do samotného vývojového procesu, kde lze ke známým problémům vždy najít doporučené řešení. Agilní metodiky, jak již bylo zmíněno, dávají přednost individualitám před procesy a nástroji. [5]

4. ONLINE REZERVAČNÍ SYSTÉM - NÁVRH

„Nejtěžší samostatnou fází stavby softwarového systému je rozhodnout, co přesně má vzniknout. Žádná z ostatních částí konceptuální práce není tak složitá, jako vybudování podrobných technických požadavků včetně všech rozhraní k lidem, strojům a dalším softwarovým systémům. Žádná z ostatních částí práce systém tak nezmrzačí, když ji uděláte špatně. Žádná z ostatních částí se tak těžko neopravuje později.“ [10]

Tato část bakalářské práce se zabývá návrhem aplikace. V praxi to znamená detailní popis všech případů užití, rolí, skupin a jejich práv, struktury aplikace, tvorbu datového modelu, návrh vzhledů jednotlivých artefaktů a ikon a mnoho dalšího. Vzhledem k omezenému rozsahu bakalářské práce zde jsou však uvedeny jen vybrané elementy aplikace, které ve fázi návrhu či vývoje byly buď klíčové pro aplikaci samotnou, nebo s sebou nesly nějaký zajímavý komplexní problém.

Aplikace slouží k rezervacím. Je to rezervační systém, který je koncipován tak, aby v budoucnu mohl být začleněn do ambulantních zařízení. Struktura aplikace počítá s tím, že na ni budou navazovat další moduly Ambulantního informačního systému. Tento modul zmíněného systému má být schopen obstarat vytvoření ambulantního zařízení, ordinace, zdravotnického personálu a pacientů v platformě UU a především zajistit vytváření objednávek.

Typově se jedná o Custom uuApp, tedy aplikace šitá zákazníkovi na míru. Typ konfigurace vychází z logiky aplikace a přání zákazníka a jde o Single-Unit aplikaci, což znamená, že je určena pro provoz s jedním operačním prostorem a s jednou konfigurací.

4.1 Zhodnocení současného stavu

Dnešní trh informačních a komunikačních technologií nabízí mnoho univerzálních informačních systémů, které slibují rychlou a snadnou implementaci. Existují ale obory specifické natolik, že při softwarovém vývoji vyžadují speciální péči a spolupráci odborníků jak ze strany softwarových profesionálů, tak ze strany odborníků v daném oboru, v tomto případě zdravotnického personálu.

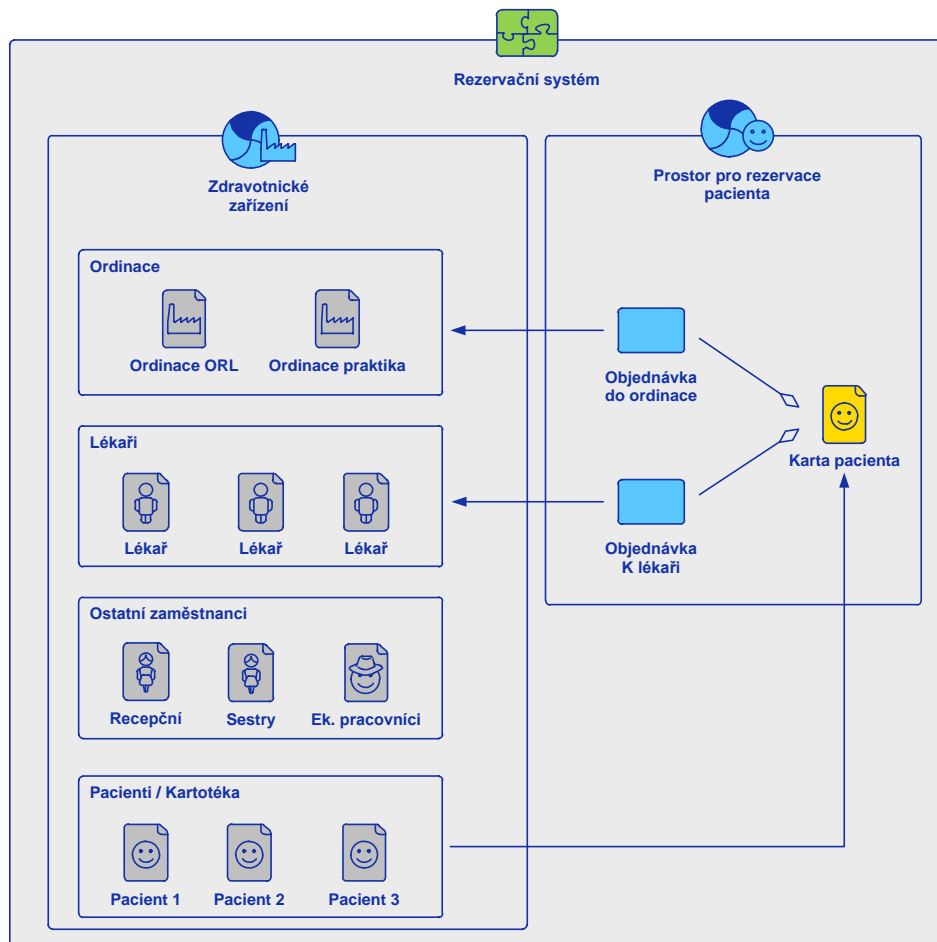
Velké nemocnice již mají v současné době kvalitní informační systémy a především si mohou dovolit vynakládat nemalé obnosy na jejich údržbu. Na druhou stranu ambulantní lékaři v mnoha případech dodnes využívají velmi zastaralé, dalo by se konstatovat až historické systémy pro správu informací (často se dokonce jedná o aplikace běžící nad MS-DOS), kde objednávání pacientů není vůbec automatizované a probíhá přes telefon.

Tato díra na trhu se pochopitelně začíná zaplňovat, a tak i společnost Unicorn nabízí svou platformu s aplikací postavenou na míru ambulantním zařízením, kterou ocení i pacienti.

4.2 Klíčová myšlenka

Tato kapitola slouží jako stručný úvod do řešené problematiky. V praxi slouží klíčová myšlenka jako zadání či podklad pro analytika potřebný ke zpracování návrhu aplikace. Jsou v ní uvedeny základní artefakty, které by v systému neměly chybět.

Online rezervační systém pro lékaře a pacienty je navržen tak, aby usnadnil práci lékařům a čas pacientům. Hlavním účelem je umožnění objednávání pacientů k lékařským prohlídkám s možností, aby byl pacient schopen se jednoduše objednat sám. Dále ovšem slouží také jako základní kámen pro budoucí, komplexnější Ambulantní informační systém určený pro lékaře.



Obrázek 7: Klíčová myšlenka rezervačního systému

Zdroj: interní dokumentace společnosti Unicorn

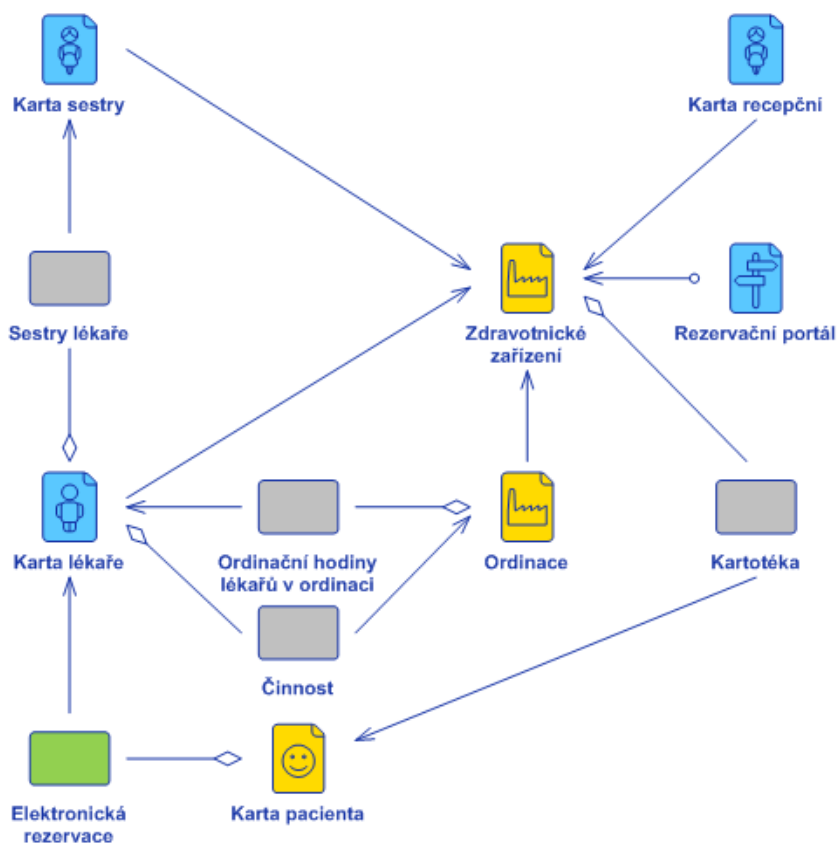
Jak je vidět na obrázku výše, **Zdravotnické zařízení** je nejdůležitější součástí rezervačního systému. V systému Unicorn Universe je tedy, stejně jako všechny ostatní objekty (krom objektů vazebních), reprezentováno artefaktem. Na Zdravotnickém zařízení se evidují zaměstnanci, ordinace a pacienti a slouží jako výchozí portál celého rezervačního systému.

Ordinace ve zdravotnickém zařízení slouží k rozřazování zaměstnanců a lékařů. Zároveň bude do budoucna sloužit k dělení financí v zařízení. Lékaři jsou nedílnou součástí ordinace, případně zařízení a pacienti mají možnost se objednat ke konkrétnímu lékaři, kterému se tato objednávka zobrazí v diáři. Ostatní zaměstnanci fungují jako podpůrní pracovníci a jejich primární úlohou je podpora základních procesů ordinace.

Karty pacientů reprezentují kartotéku. Kartotéka schraňuje informace o jednotlivých pacientech a platí, že každý pacient je opět reprezentován artefaktem. Vedou se o nich záznamy jak osobní, tak lékařské, případně i seznam návštěv, kdy se pacient v rámci rezervačního systému objednal. Artefakt Pacient (v nákresu žlutě zvýrazněn) umístěný v prostoru pro rezervace pacienta slouží právě jen k provedení rezervace a obsahuje pouze údaje nutné pro uskutečnění této rezervace.

4.3 Produktový pohled

Produktový pohled, na rozdíl od klíčové myšlenky, již obsahuje všechny artefakty, které budou v rezervačním systému hrát nějakou roli. Zároveň jsou zde znázorněny i vazby mezi těmito artefakty, které jsou pro vývoj aplikace, datový model a vůbec pro celou business logiku řešené problematiky důležité.



Obrázek 8: Znázornění vazeb mezi objekty systému

Zdroj: interní dokumentace společnosti Unicorn

Zelenou barvou je znázorněn nejdůležitější objekt, kterým je Elektronická rezervace. V systému není reprezentován artefaktem, nýbrž je to ‘pouze‘ vztah mezi dvěma jinými artefakty uložený v příslušném schématu databáze jako vazební objekt (tj. objekt bez unikátního data klíče, místo kterého je dohledatelný přes dva cizí klíče – klíč Karty pacienta a Karty lékaře).

Pro systém rezervací jsou nejdůležitějším zdrojem dat artefakty Karta pacienta, Karta lékaře a Ordinance.

4.4 Analýza případů užití

Na samotném začátku návrhu rezervačního systému, nebo obecně jakékoli jiné aplikace, je potřeba se zamyslet nad nároky, které budou jednotlivými uživateli na systém kladeny. Výsledkem této analýzy by měl být popis funkcionality systému tak, jak ho uvidí uživatel. Zároveň tato část dokumentace slouží jako zadání pro vývojáře, nicméně jejím cílem není sloužit jako návod pro samotné technické řešení, to zůstává v kompetenci vývojáře.

Nejdůležitějšími výstupy této analýzy pro potřeby návrhu aplikací na platformě Unicorn Universe jsou nalezení aktérů a nalezení případů užití.

4.4.1 Nalezení aktérů

Dalším přínosem analýzy případů užití je nalezení aktérů (v tomto případě rolí a skupin), které budou mít v aplikaci specifická práva. Toto nalezení zpravidla probíhá ve dvou krocích. V prvním kroku je potřeba se zamyslet, které role jsou typické pro danou business logiku a jsou vzájemně odlišné natolik, že potřebují být reprezentovány vlastním artefaktem.

Druhým krokem je doplnění skupin a rolí z prvního kroku o standardní role, které by měly být v každé organizační jednotce. Jedná se o roli Authority (jelikož systém organizačních jednotek reprezentuje reálnou organizační strukturu podniku, role Authority je zpravidla vedoucí dané jednotky) a o skupiny Authorities, Executives a Auditors. Druhý krok je možné vynechat, pokud je z prvního kroku dostatek rolí a skupin na pokrytí standardu z kroku druhého. Výsledkem zmíněného uvažování jsou následující skupiny:

- **Hlavní lékaři zařízení** – role s právem vytvářet nové uživatele a ordinace a upravovat zdravotnické zařízení.
- **Hlavní lékaři ordinace** - role s právem vytvářet nové uživatele a ordinace a upravovat zdravotnické zařízení (zahrnuje méně případů užití než Hlavní lékaři zařízení).
- **Lékaři zařízení** – role s právem vytvářet nové uživatele (kromě hlavního lékaře), úpravy ordinace a úpravy vlastního profilu.
- **Lékaři ordinace** - role s právem vytvářet nové uživatele (kromě hlavního lékaře), úpravy ordinace a úpravy vlastního profilu (zahrnuje méně případů užití než Lékaři zařízení)
- **Sestry zařízení** – role s právem vytvářet a objednávat pacienty.
- **Sestry ordinace** – role s právem vytvářet nové pacienty.
- **Recepční zařízení** – role s právem vytvářet a objednávat pacienty.
- **Recepční ordinace** – role s právem zobrazit rezervační systém.

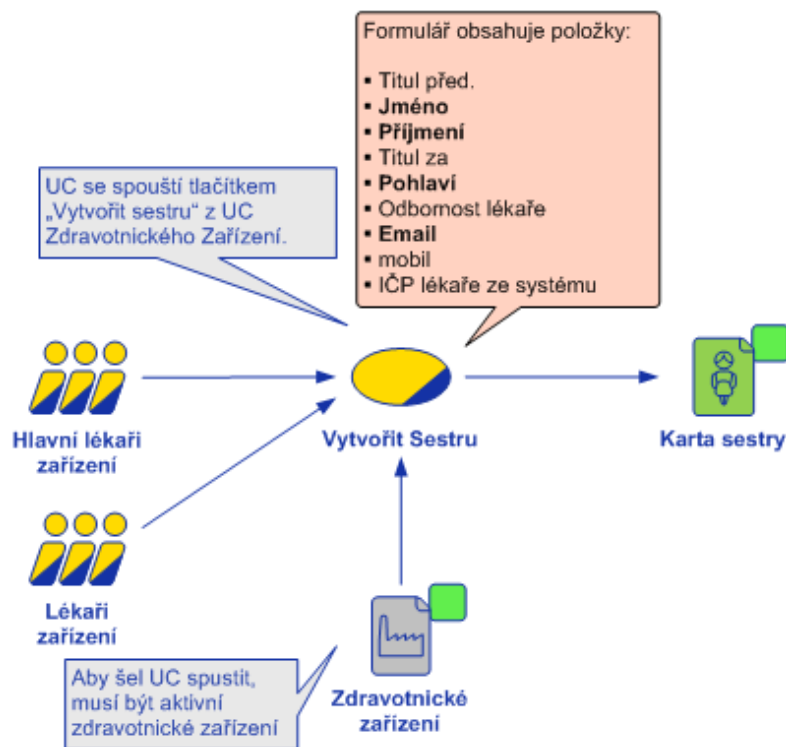
4.4.2 Nalezení případů užití

Na rozdíl od klasické analýzy případů užití, kterou využívá většina IT světa, vizualizační jazyk uuBML používaný k návrhu aplikací má svou vlastní reprezentaci, a co do rozsahu celé dokumentace, tvoří tyto nákresy převážnou část. Od klasické analýzy případů užití se liší především v tom, že každý případ užití je realizován vlastním nákresem a tento nákres obsahuje více informací. Dále je textem, který ve značné míře duplikuje nákres, ale zároveň obsahuje alternativní scénáře.

V době, kdy je návrh aplikace ve fázi analýzy případů užití, by již měl být vypracovaný produktový pohled. Každá aplikace totiž obsahuje určité typy funkcí, které jsou běžné pro každou aplikaci a v jejichž průběhu dochází k úpravě vazeb mezi artefakty. Standardně se jedná o funkce, které zakládají nějaký artefakt (těmi je dobré v návrhu začínat), dále které upravují jeho vlastnosti anebo jej ruší.

4.4.2.1 Případ užití – Vytvořit sestru

Pro demonstraci uplatnění této analýzy je uveden základní případ užití, který zakládá artefakt a má své vstupní a výstupní podmínky.



Obrázek 9: Ukázka popisu případu užití – Vytvořit sestru

Zdroj: interní dokumentace společnosti Unicorn

Na nákrese (viz obrázek 9) je zobrazena uuBML reprezentace případu užití Vytvořit sestru. Ikony rolí vlevo znázorňují role oprávněné tuto funkčnost spouštět. Ikona artefaktu Zdravotnické zařízení se zeleným políčkem znázorňující aktivní stav udává podmínku, která pro spuštění dané funkčnosti musí být splněna. To, že musí být nějaký artefakt v aktivním stavu, je poměrně běžná podmínka – nejen že je z tohoto artefaktu funkčnost spouštěna, ale zároveň, jak je vidět v produktovém pohledu, dochází k vytvoření vazby mezi Zdravotnickým zařízením a nově vytvořenou Kartou sestry.

Jelikož se jedná o případ užití, který je navenek reprezentován jako formulář, pole tohoto formuláře jsou popsány v červené bublině (tučně zvýrazněný text určuje povinné atributy). Zelenou barvou je opět zvýrazněna nejdůležitější entita nákrese, tedy nově vytvořená Karta sestry v aktivním stavu. Šedé bubliny poskytují doplňující informace.

4.4.2.2 Případ užití – Naplánovat prohlídku

Ne všechny případy užití lze ale znázornit pomocí jazyka uuBML. Výjimkou jsou především případy užití, které obsluhují widget. Jelikož systém rezervací je řešen pomocí widgetu a případy užití s ním spojené nevytváří ani neruší žádné artefakty ani vazby mezi nimi, nýbrž obsluhují své vlastní schémata v databázi, není funkčnost tohoto případu užití vizuálně znázorněna.

Případ užití Naplánovat prohlídku se může zprvu jevit jako elementární funkčnost nebo jako výchozí případ užití celé aplikace, kterým by každý návrhář začal svou analýzu. Výchozí funkčností by se sice nazvat dal, ale elementárním tento případ užití jistě není. Jelikož rezervace samotné jsou v systému reprezentovány widgetem a jejich data musejí být konzistentně zobrazená jak ve widgetu, tak v systému UU, nastává zde optimalizační problém.

Každou rezervaci si zadavatel aplikace přál zobrazit v diáři lékaře a pacienta, což znamená, že s každou naplánovanou prohlídkou se musí vytvořit aktivita typu schůzka. Zároveň je ale potřeba, aby byla aplikace schopná v těchto aktivitách rychle vyhledávat a hledat volná místa při plánování další prohlídky. Tento požadavek má za následek nutnost ukládat jednotlivé rezervace i do databáze Object Store. Samotné vyhledávání volného termínu vyžaduje vhodný algoritmus, který dokáže mezi rezervacemi nalézt vhodnou časovou pauzu v rámci ordinačních hodin lékaře a to tak, aby nedocházelo k nadměrnému stahování dat.

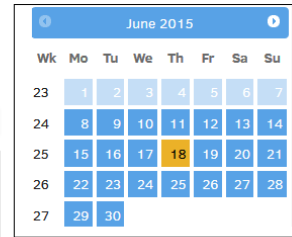
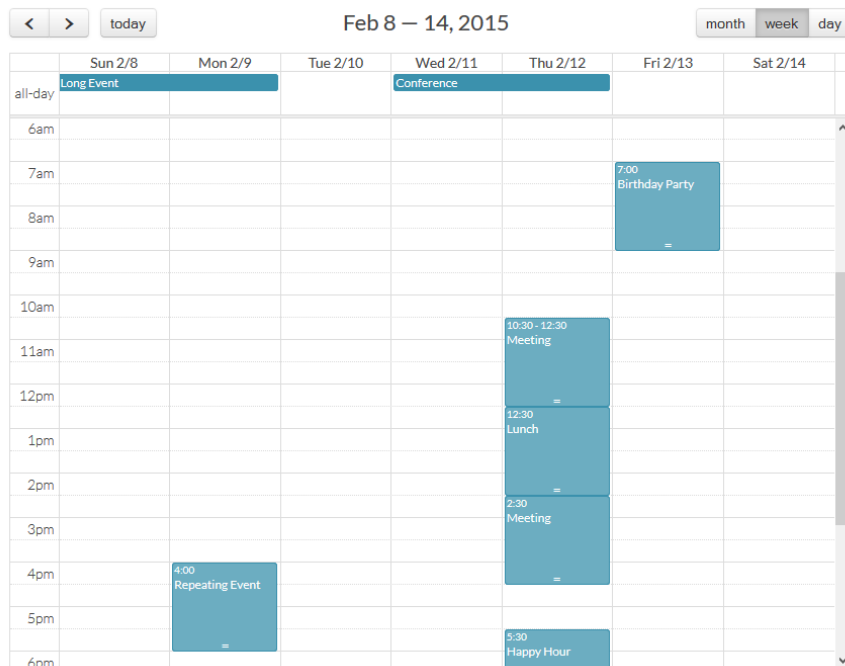
4.5 Návrh rezervačního widgetu

Jelikož funkčnost vyžadovaná od rezervačního systému je natolik komplexní a specifická a za pomocí VUCů by její implementace byla velmi neobratná, proběhne její realizace za pomoci technologie widgetů. Tento widget tedy nebude složen z předpřipravených komponent, ale bude umístěn uvnitř popisu artefaktu a bude schopen s platformou UU komunikovat.

Hlavními požadavky na widget byla funkcionalita a intuitivní používání, aby nebylo třeba jeho uživatele školit k používání aplikace. Navíc by měl widget umožnit jednoduchý výběr ordinace, lékaře a druhu zákroku. Při vytváření objednávky bude pacient informován o standardní délce onoho zákroku.

MUDr. Jan Novák v zařízení Clínica Bohemica a.s.

Zvolte termín na který se chcete objednat ->



Pro vybrání termínu, klikněte do prázdného pole, kde je volno pro Vaši objednávku.



Povybrání volného termínu se zobrazí toto dialogové okno:

Pacient: Petr Straka - objednávka rezervace do ordinace na Hájku k lékaři : MUDr. Jan Novák

Lékař: MUDr. Jan Novák

Ordinace: Ordinace na Hájku

Date: 2015-06-19

From - To: 13:30 - 14:00 (00:30)

Činnost: Prevence

Poznámka lékaři: Přijdu včas

User Name: Milan Brašna (12-283-1)

OK Storno

Nepracovní doba se v kalendáři bude zobrazovat šedě, stejně jako polední pauza. Případně plné termíny v kalendáři.

Víkendy se v kalendáři zobrazovat nebudou.

Zde by mohlo být nastavení notifikací ohledně objednávky, nebo to už máme jinde ? není to zde nutné.

Pro potvrzení objednávky klikněte na OK.

Obrázek 10: Návrh rezervačního widgetu

Zdroj: interní dokumentace společnosti Unicorn

5. ONLINE REZERVAČNÍ SYSTÉM - REALIZACE

V době vypracovávání této bakalářské práce se aplikace nachází ve fázi vývoje a interního testování, nicméně po osmnácti stech odpracovaných hodinách a několika kontrolních dnech, prezentací dosavadních výsledku zadavateli a změnových požadavcích je již velká část aplikace hotova a otestována.

5.1 Naplánovat prohlídku

Při návrhu aplikace a jejím následném vývoji může nastat situace, kdy v návrhu došlo k nesprávnému odhadu pracnosti. Z pohledu designéra se může nějaký velmi komplexní a technologicky náročný problém zdát triviálním, i když tomu tak ve skutečnosti není. To může mít za následek nedostatečnou alokaci zdrojů na vývoj dané aplikace.

V případě případu užití Naplánovat prohlídku došlo ale k opačnému stavu. Tato funkčnost sice je velmi komplexní a vyžaduje bezchybnou souhru několika technologií, ale její implementace byla v konečném důsledku oproti návrhu relativně snadná. Žádný složitý algoritmus na vyhledávání vhodných oken rezervací nakonec nebylo potřeba vymýšlet. Namísto toho se přistoupilo k jednoduššímu a více uživatelsky přátelskému řešení.

Celá funkčnost je rozdělena na synchronní a asynchronní část. Synchronní část zajistí jen kontrolu volného času a uložení objektu s rezervací do databáze. Rezervační objekt nemá vlastní data klíč a distribuuje pouze třemi atributy – čas od, čas do a atribut *in_creation*, který zajistí neprůstřelný souběh paralelního spuštění funkčnosti (mohlo by se stát, že uživatel zruší rezervaci dříve, než se dokončí asynchronní část algoritmu).

Po dokončení synchronní části, jejíž délka nepřesahuje pár sekund, dojde k zavolání makra, které již pracuje na pozadí. Toto makro vytvoří aktivitu, která se zobrazí doktorovi a pacientovi v diáři, dále skript naplánuje notifikace a opět dojde k upravení objektu rezervace v databázi.

Výhodou takového řešení je především to, že k zjištění volných termínů nedochází přes systémový komand (který by mohl trvat několik sekund), ale bohatě postačí jeden dotaz do databáze. Díky tomu, že zdravotní sestra má právo zobrazit widget a rezervovat pacienta, je tak schopná se v kalendáři podívat, kam potřebuje a ústně s pacientem domluvit vhodný termín.

Naplánovat návštěvu u lékaře

Vyberte si prosím ordinaci nebo lékaře z našeho zdravotnického zařízení a dále pokračujte výběrem činnosti na kterou se chcete objednat.

Dermatovernerologická amb... MUDr. Petr Kožní Čištění kožních skvrn

Nyní Měsíc Týden Den 28 březen — 3 duben 2016

po 28.3 út 29.3 st 30.3 čt 31.3 pá 1.4 so 2.4 ne 3.4

Nová rezervace

Činnost Čištění kožních skvrn
Délka činnosti 120 minut
Ordinace Dermatovernerologická ambulance
Lékař MUDr. Petr Kožní
Datum 31.03.2016
Od - Do 09:30 - 11:30
Jméno pacienta
Poznámka

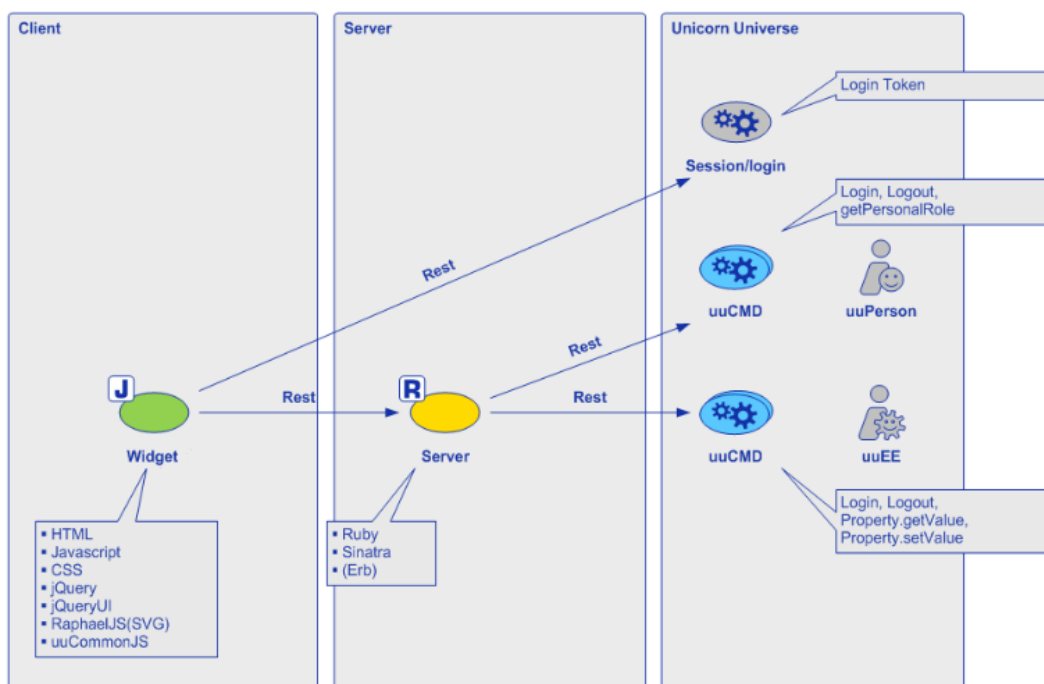
OK Storno

Obrázek 11: Realizace případu užití – Naplánovat prohlídku

Zdroj: Aplikace Plus4U Lékař

5.2 Rezervační widget

Rezervační widget byl implementován pomocí třívrstvé architektury, která se v UU pro widgety běžně používá. Jedná se o klientskou část, serverovou část a jako třetí vrstva je chápána celá platforma UU, z níž widget přebírá veškeré funkčnosti. Do popisu artefaktu se pomocí BB kódu vloží komponenta typu widget, která umožní skriptům z klientské části změnit vzhled a rozšířit funkčnost platformy.



Obrázek 12: Třívrstvá architektura rezervačního widgetu

Zdroj: interní dokumentace společnosti Unicorn

V klientské části widgetu se pracuje s jazykem Javascript, jenž umožňuje vytvářet interaktivní prvky grafického rozhraní, animace a efekty obrázků. Tato vrstva widgetu tedy obsahuje mnoho funkcí, které společně tvoří uživatelské rozhraní mini aplikace. Zároveň však obsahuje funkce, které pomocí rozhraní Rest volají serverové funkce.

Technologie tvořící serverovou část widgetu zahrnují programovací jazyk Ruby a k postavení samotného serveru je využito gemu (knihovny) Sinatra. Standardní funkcí, která se provede vždy při aktualizaci stránky s komponentou widget, je volání metody getContent, jenž klientské části poskytne základní data pro zobrazení uživateli. Dále jsou volány metody potřebné pro vytvoření, rušení a úpravu rezervací.

Na jednom z kontrolních dnů aplikace ovšem došlo k vznesení požadavků na face-lift celého widgetu. Díky třívrstvé architektuře widgetu úprava takového typu ovlivňuje pouze klientskou část, a serverová část tak může zůstat beze změny. Proto proběhla nová mini etapa návrhu vzhledu widgetu, na jejíž implementaci se v současné době pracuje a jejíž ukázkou lze nalézt v příloze této práce (viz příloha A).

5.3 Zhodnocení přínosu

Vzhledem k faktu, že v době zpracovávání této bakalářské práce je zkoumaná aplikace stále ve fázi vývoje, stěží lze posoudit její celkový přínos. Nicméně během celé fáze návrhu a implementace byl kladen velký důraz na uživatelskou přívětivost aplikace a možnost samoobsluhy, která s tím jde ruku v ruce. Všechny informace, co má daný uživatel vidět, jsou dobře viditelné a intuitivně umístěné. Všechny nedostatky, které odhaluje testování aplikace implementačním i akceptačním týmem, jsou postupně zapracovávány.

Největším přínosem je ovšem bezpochyby rezervační widget, jenž je nyní už funkční a probíhají na něm pouze kosmetické úpravy. Widget je solidně zpracovaný a objednávání pacientem v něm probíhá velmi rychle a přehledně. V porovnání s procesem vytváření objednávek k obvodním lékařům, zubařům a jiným ambulantním klinikám je řešení nabízené aplikací Plus4u Lékař opravdu velmi pohodlné a rychlé.

ZÁVĚR

Hlavním cílem této bakalářské práce byl návrh a realizace online rezervačního systému pro lékaře a pacienty na platformě Unicorn Universe. Teoretická část této práce pokrývá základní vlastnosti platformy UU, jejichž znalost je klíčová pro plné pochopení návrhu a realizace dané aplikace.

Na aplikaci online rezervací byly od počátku kladeny dva primární požadavky. Prvním bylo co nejvíce usnadnit práci lékařům a čas pacientům s možností, aby byl pacient schopen se k lékaři objednat sám z pohodlí svého domova. Tohoto cíle bylo dosaženo jednak díky rozhodnutí, že systém rezervací bude realizován přes technologii widgetu, a následně díky úspěšné implementaci. Druhým cílem aplikace byla schopnost sloužit jako stavební kámen budoucího komplexnějšího ambulantního systému určeného pro lékaře. Úspěšnost zmíněného milníku se v současné době stěží posuzuje, jelikož aplikace online rezervací je stále ve vývoji, nicméně první návrhy následujících modulů jsou již zpracované, takže se od tohoto cíle neupouští a bude se pracovat na jeho splnění.

Úplným závěrem bych rád vyjádřil své sympatie k aplikaci Plus4U Lékař. Rád bych tento (nebo alespoň obdobný) systém online rezervací k lékaři jako pacient využíval v každodenním životě, a nemusel tak procházet klasickým domlouváním termínu návštěvy přes email a předešel tak neefektivně využitému času nebo ještě hůř, několika hodinám stráveným v čekárně.

SEZNAM POUŽITÉ LITERATURY

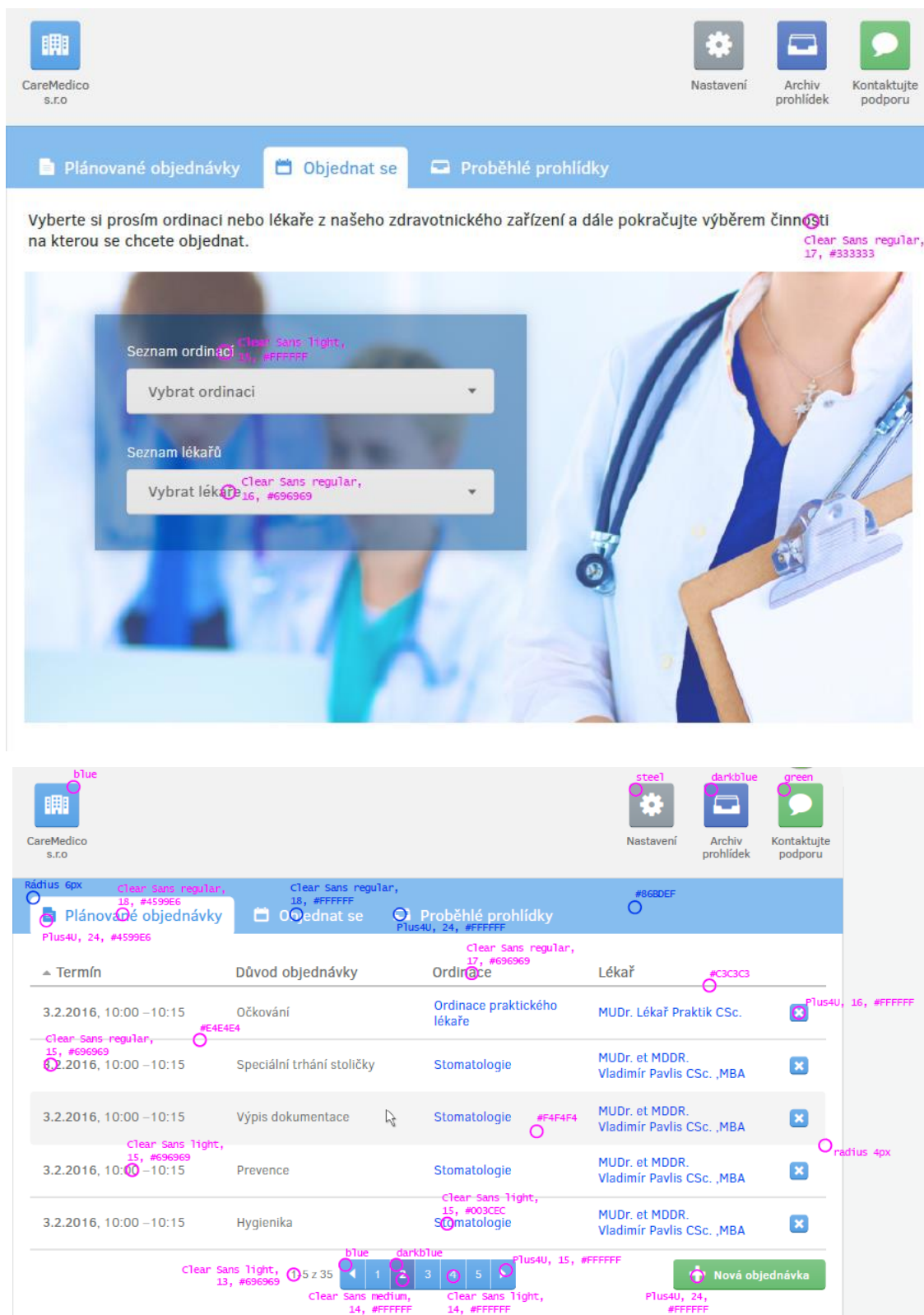
- [1] KOVÁŘ, V. *Metodika Unicorn Enterprise System Powered Compan: Metodika pro řízení podniku a organizací s přímou podporou informačního systému*. 2011. 130 s. Dizertační práce. Univerzita Hradec Králové, Fakulta informatiky a managementu, Katedra informatiky a kvalitativních metod. Unicorn Universe [online]. 2012 [cit. 2016-04-03]. Dostupné z: <https://unicornuniverse.eu>
- [2] UNICORN A.S. *Unicorn Universe*. [online]. [vid. 2016-04-03]. Dostupné z: <https://unicornuniverse.eu>
- [3] FROST, Raymond & PIKE, Jacqueline. *Business information system: design an app for that*. Washington, DC: Flat World Knowledge, 2013 [cit. 2016-04-03]. ISBN 978-1-4533115-7-8
- [4] PADADOPOULOS, Georgios. *Moving from traditional to agile software development methodologies also on large, distributed projects* [online]. 2014 Dostupné z <http://www.sciencedirect.com/science/article/pii/S1877042815012835>
- [5] Skupina autorů. *Manifest agilního vývoje* [online]. Dostupné z: <http://agilemanifesto.org/iso/cs/>
- [6] MOE, Nils & DINGSOYR, Torgeir & DYBA, Tore. *A Teamwork model for understanding an agile team: A case study of a Scrum project* [online]. 2008 Dostupné z: <http://www.sciencedirect.com/science/article/pii/S0950584909002043>
- [7] ALMEIDA, Maria. *How to be a great Product Owner* [online]. 2015 Dostupné z: <http://blog.landing.jobs/how-to-be-a-great-product-owner/>


- [8] PICHLER, Roman. *Agile product management with Scrum: creating products that customers love*. Upper Saddle River, NJ: Addison-Wesley, c2010. Addison-Wesley signature series. ISBN 0321605780.
- [9] STRAY, Viktoria; SJOBERG, Dag; DYBA, Tore. *The daily stand-up meeting: A grounded theory study* [online]. 2016 University of Oslo. Dostupné z: <http://www.sciencedirect.com/science/article/pii/S0164121216000066>
- [10] WIEGERS, Karl Eugene. *Požadavky na software*. Vyd. 1. Brno: Computer Press, c2008. ISBN 978-80-251-1877-1. [cit. 2016-04-03].

SEZNAM PŘÍLOH




Příloha A: Ukázka návrhu nového vzhledu rezervačního widgetu	46
--	----

Příloha A: Ukázka návrhu nového vzhledu rezervačního widgetu





CareMedico
s.r.o.

Nastavení Archiv prohlídek Kontaktujte podporu

Plánované objednávky
Objednat se
Proběhlé prohlídky

Vyberte si prosím ortopedii na kterou se chcete objednat


Ortopedie

Nyní

Po 1

8:00	8:00-8:20				
9:00	9:00-9:50				
10:00	10:00-10:20				
11:00			11:00-12:00		
12:00	12:05-14:35	12:00-12:20			12:00-12:50
			12:30-12:50		
13:00			13:05-14:20	13:10-13:45	

Nová rezervace



MUDr. Ondřej Klicman
Gynekologická ordinace

Plus4U, 22, #696969

Prevence
30 minut

Datum 11.3.2016

Od - Do 16:30 - 17:00

Jméno pacienta Milan Brašer

Poznámka

OK
storno

Činnosti

13 březen 2016

5

stín #00000, krycí #88