

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Zimmermannová** Jméno: **Michaela** Osobní číslo: **485091**
Fakulta/ústav: **Fakulta informačních technologií**
Zadávající katedra/ústav:
Studijní program: **Informatika**
Studijní obor: **Webové a softwarové inženýrství**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Asterion - frontend vizualizace časových os

Název bakalářské práce anglicky:

Asterion - timelines visualization frontend

Pokyny pro vypracování:

Asterion je rozsáhlý svět s bohatou a komplikovanou historií. Cílem práce je vizualizovat historická data tohoto fiktivního světa pomocí vícera časových os vázaných na konkrétní téma, lokace, apod. Výsledný produkt bude běžet jako webová služba. Práce je součástí skupinového projektu Asterion.

- 1) Proveďte rešerši způsobů vizualizace historických dat.
- 2) Analyzujte a vhodně uspořádejte množinu časových dat světa Asterion poskytnutých zadavatelem.
- 3) Proveďte rešerši technologického řešení webové vizualizace.
- 3) Navrhněte prototyp - zaměřte se na frontend, především se soustřeďte na vzhled, UI, uživatelskou přehlednost atp.
- 4) Implementujte a nasadte službu.
- 5) Proveďte důsledné uživatelské testování.

Seznam doporučené literatury:

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Radek Richtr, Ph.D., katedra softwarového inženýrství FIT

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **16.02.2021** Termín odevzdání bakalářské práce: _____

Platnost zadání bakalářské práce: _____

Ing. Radek Richtr, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Studentka bere na vědomí, že je povinna vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studentky

Bakalářská práce

ASTERION - FRONTEND VI- ZUALIZACE ČASOVÝCH OS

Michaela Zimmermannová

Fakulta informačních technologií ČVUT v Praze
Katedra softwarového inženýrství
Vedoucí: Ing. Radek Richtř, Ph.D.
27. června 2021

České vysoké učení technické v Praze
Fakulta informačních technologií

© 2020 Michaela Zimmermannová. Všechna práva vyhrazena.

Tato práce vznikla jako školní díla na Českém vysokém učení technické 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 bez uplatněných zákonných licencí nad rámec oprávnění uvedených v Prohlášení je nezbytný souhlas autora.

Odkaz na tuto práci: Michaela Zimmermannová. *Asterion - frontend vizualizace časových os*. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2021.

Obsah

Poděkování	vii
Prohlášení	viii
Abstrakt	ix
Shrnutí	x
Seznam zkratek	xi
1 Rešerše způsobů vizualizace historických událostí	1
1.1 Časová osa	1
1.2 Způsoby vizualizace	1
1.3 Použité pojmy	3
1.4 Příklady způsobů vizualizace	3
1.4.1 World Anvil	3
1.4.2 Timeline Maker Pro	3
1.4.3 Timetoast	4
1.4.4 TimelineJS	6
1.4.5 Tiki-Toki	7
2 Analýza podobných aplikací	9
2.1 World Anvil	9
2.2 TimelineJS	10
2.3 Timeline Maker Pro	10
2.4 Tiki-Toki	10
2.5 Timetoast	11
2.6 Závěr	11
3 Rešerše technologií	13
3.1 HTML, CSS a JS	13
3.2 Canvas, WebGL a SVG	13
3.3 Knihovny pro 2D grafiku	14
3.4 Závěr	14
4 Historická data Asterionu	15
4.1 Prostředí Asterionu	15
4.1.1 Kalendářní systém	15
4.2 Rozdělení dat na entity	15
4.2.1 Události	17
4.2.2 Tagy	17
4.3 Metodika tvoření entit	17

5	Návrh webové aplikace	19
5.1	Návrh stránky Asterion	19
5.1.1	Hlavička stránky	19
5.1.2	Zobrazení časové osy	19
5.1.3	Vzhled časové osy	20
5.1.4	Událost	20
5.1.5	Popisek časové osy	20
5.1.6	Rozhraní pro ovládání vizualizace	21
5.2	Stránka pro administrátory	23
6	Softwarový návrh	25
6.1	Funkční Požadavky	25
6.1.1	Zobrazení událostí na časové osy	25
6.1.2	Nastavení vizualizace časových os	25
6.1.3	Výběr událostí k vizualizaci	25
6.1.4	Evidence událostí, tagů, aliasů a kategorií	25
6.1.5	Správa administrátorů	26
6.1.6	Autorizace a autentizace administrátorů	26
6.2	Nefunkční požadavky	26
6.2.1	Webová aplikace	26
6.2.2	Grafické uživatelské rozhraní	26
6.3	Případy užití	26
6.3.1	Výběr tagu	26
6.3.2	Změna zobrazeného časového období	26
6.3.3	Prohlížení popisu události	26
6.3.4	Upravení pořadí a počtu časových os	26
6.3.5	Změna barvy časové osy	27
6.3.6	Přidání nových dat	27
6.3.7	Úprava dat	27
6.3.8	Odebrání dat	28
6.3.9	Správa administrátorů	28
6.4	Doménový model	28
6.4.1	Alias	29
6.5	Diagram tříd	29
6.5.1	Data	30
6.5.2	Api	30
6.5.3	Vizualizace	30
6.5.4	Editor	33
6.5.5	Pomocné třídy	33
7	Implementace	35
7.1	Použité technologie	35
7.2	První verze	35
7.3	Druhá verze	36
7.4	Aktuální verze	36
8	Uživatelské testování	37
8.1	Cílová skupina	37
8.2	Návrh úkolů	37
8.3	Výsledek testování	37
9	Závěr	39

A Formulář pro uživatelské testování	41
--------------------------------------	----

Obsah přiloženého média	49
-------------------------	----

Seznam obrázků

1.1	Časová osa spirálového typu v aplikaci Ross Spiral Curriculum	2
1.2	Část časové osy ve World Anvil	4
1.3	Časová osa v Timeline Maker Pro, zobrazení Flag	5
1.4	Část časové osy v Timetoast v režimu Timeline	5
1.5	Část časové osy v Timetoast režimu List	6
1.6	Časová osa ve TimelineJS	6
1.7	Časová osa v aplikaci Tiki Toki ve 2D režimu	7
1.8	Časová osa v aplikaci Tiki Toki ve 3D režimu	8
4.1	Mapa Lendoru	16
4.2	Mapa Tary	16
4.3	Ukázka z kroniky Asterionu	17
5.1	Návrh hlavičky webové aplikace	19
5.2	Návrh časové osy	20
5.3	Návrh událostí	20
5.4	Návrh popisku časové osy	21
5.5	Návrh ovládacích prvků v kontrolním panelu	21
5.6	Návrh okna pro výběr barvy	22
6.1	Případy použití aplikace Asterion timelines	27
6.2	Případy použití Editoru	28
6.3	Doménový model	29
6.4	Model tříd nesoucí data	30
6.5	Model tříd pro komunikaci se serverem	31
6.6	Model tříd tvořící základní strukturu programu	32
6.7	Základ modelu tříd editoru	33
6.8	Model pomocných tříd	34
7.1	První verze prototypu	36

Chtěla bych poděkovat především svému vedoucímu práce Ing. Radkovi Richtrovi Ph. D. za cenné rady, věcné připomínky a vstřícnost při konzultacích a vypracování bakalářské práce. Dále bych chtěla poděkovat Tomášovi Růžičkovi za spolupráci na tvorbě této práce. V poslední řadě bych ráda poděkovala své rodině a přátelům za podporu, víru a trpělivost.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů 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 předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výtěžným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené.

V Praze dne 10. května 2020

.....

Abstrakt

Tato práce se zabývá vizualizací časových událostí Asterionu na časové osy. Asterion je český fantasy svět pro hraní her na hrdiny, jehož historie je velmi spletitá a vizualizace si dává za cíl zjednodušit orientaci v těchto datech. Cílem práce bylo vytvořit webovou stránku, kde bude snadné získat přehled o sledu událostí pomocí vyhledání klíčového slova a následného zobrazení událostí týkajících se tohoto klíče na časovou osu. Sekundárním cílem práce bylo vytvořit webovou stránku, kde by bylo možné data spravovat. Před samotnou tvorbou webových stránek byla provedena rešerše a analýza aplikací, které se zabývají zobrazením časových os, které napomohly návrhu webové aplikace. Při implementaci byla použita technologie React a pro vykreslení vizualizace knihovna PIXIJS. Na závěr bylo provedeno uživatelské testování obou webových stránek.

Klíčová slova vizualizace historických událostí, časové osy, webová aplikace, návrh uživatelského rozhraní React, PIXI.js

Abstract

This work deals with the visualization of time events of Asterion on timelines. Asterion is a Czech fantasy world for role-playing games whose history is very complicated, and the visualization aims to simplify orientation in this data. The work aimed to create a website where it will be easy to get an overview of the sequence of events by searching for a keyword and then displaying events related to this keyword on the timeline. The secondary goal of the work was to create a website where it would be possible to manage the data. Before the creation of the website, a search and an analysis of applications that deal with the visualization of timelines was performed to help design the application. React technology was used during the implementation and the PIXIJS library was used to render the visualization. Finally, user testing of both websites was performed.

Keywords visualization of historical events, timelines, web application, user interface design, React, PIXI.js

Shrnutí

Motivace

Aplikace, která bude výsledkem této práce, bude sloužit komunitě hráčů Dračího doupěte ve fiktivním světě Asterionu.

Dračí doupě je česká fantasy hra na hrdiny od vydavatelství Altar, která částečně vychází z americké hry Dungeons & Dragons. Během hry hráči zaujímají role fiktivních postav a účastní se společného příběhu, který jim dopředu připraví (vymyslí, či převezme) jeden z hráčů, který je označován jako Pán jeskyně. Pán jeskyně se stává vypravěčem příběhu a řídí chod hry. Hra se obvykle odehrává v některém z fiktivních fantasy světů, mezi něž patří i svět Asterionu.

Asterion je svět, který byl vytvořen v roce 1999 od té doby získal velmi rozsáhlou historii, kterou tvoří 13 herních modulů a více než 10 knih. Kvůli složitosti a komplexnosti této historie není pro hráče a pro spisovatele snadné orientovat se v událostech Asterionu a vytvářet nové příběhy bez porušení již existujících příběhových linií.

Cíl práce

Cílem práce je navrhnout, implementovat a otestovat frontend webové aplikace, která bude přehledně vizualizovat historické události Asterionu a usnadní jejich vyhledávání. Tato aplikace bude také zahrnovat editor, který bude umožňovat administrátorům úpravu dat.

Související práce

Práce bude navazovat na prototyp aplikace, která vznikla v rámci předmětů BI-SP1 a BI-SP2 ve spolupráci s Janem Krkoškou, Ondřejem Švejstílem, Kryštofem Dostálem a Tomášem Růžičkou. Současně s tímto projektem, který se soustředí na frontend webové aplikace, vzniká bakalářská práce zaměřená na backend této aplikace. Jejím autorem je Tomáš Růžička.

Seznam zkratek

HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
JS	Javascript
WebGL	Web Graphics Library
API	Application Programming Interface

Rešerše způsobů vizualizace historických událostí

V této kapitole krátce popíši, co je to časová osa, rozdělím časové osy do několika kategorií podle tvaru časových os. Vysvětlím některé z mnoha používaných pojmů, které budu používat i v následujících kapitolách a popíši některé z aplikací, které se zabývají vizualizací časových os.

1.1 Časová osa

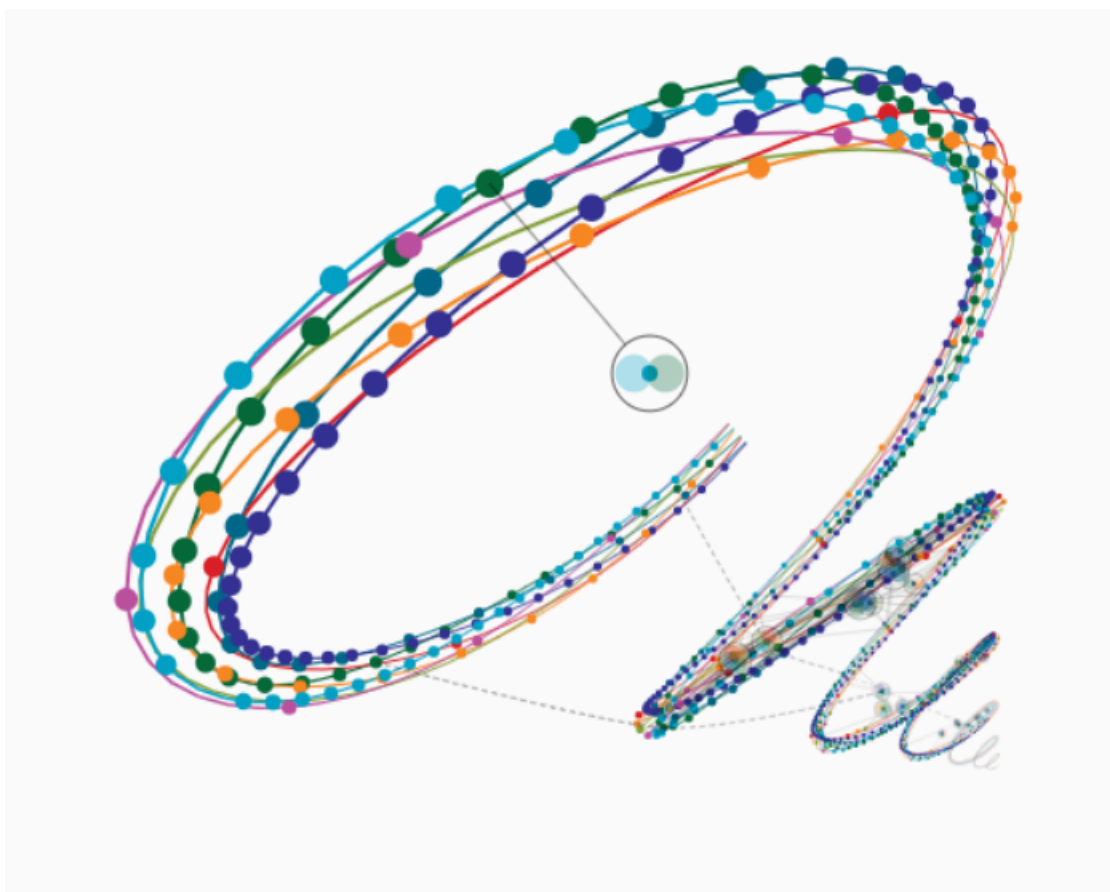
Dle M. Brehmer a spol. [1]

časová osa zobrazuje sled událostí. Jednoduchá osa označuje typy zobrazovaných událostí, jejich počet a pořadí, ve kterém k nim došlo. Podrobnější časová osa může indikovat, jak dlouho trvaly a zda se některé z událostí překrývaly. Událost je obvykle zakódována pomocí nějaké grafické značky, například čáry nebo ikony. Umístění této značky ve vztahu k ose představující čas a k dalším značkám událostí indikuje, kdy k událostem došlo.

1.2 Způsoby vizualizace

M. Brehmer a spol. [1] rozdělují možné reprezentace časové osy podle jejího tvaru. Nejpoužívanějším typem je lineární reprezentace, která zobrazuje události na jednu nebo více přímkách, nejčastěji orientovaných zleva doprava, ale často také shora dolů. Více časových os umožňuje rozdělit zobrazované události podle nějaké metriky do kategorií a se správnou granularitou přehlednit jejich zobrazení. Dalším možným typem je paprskovitá reprezentace, která umožňuje zaznamenat periodické události, např. životní cyklus člověka. Ukázkovým příkladem je *Creative routines* [2], které popisuje denní rituály historických osobností. Dalším způsobem vizualizace je mřížka. Jedná se o vsudypřítomný typ, pod jehož názvem se skrývá klasický kalendář. Spirálový typ má tvar spirály a díky tomuto tvaru umožňuje prezentovat velké množství událostí na jedné ose. Příkladem této reprezentace je *Ross Spiral Curriculum* [3], které zobrazuje vývoj člověka interaktivním způsobem na 3D spirále a je zobrazen na obrázku 1.1.

Posledním typem jsou různé reprezentace, které vypadají jako křivky nebo bodové diagramy. Stejně jako spirálové reprezentace, jsou vhodné pro zobrazování velkého množství událostí a jsou často používány ke vzdělávacím účelům. Příklad této reprezentace je *NBA Lockout* [4].



■ **Obrázek 1.1** Časová osa spirálového typu v aplikaci Ross Spiral Curriculum

1.3 Použité pojmy

- **Trvající událost** je událost, která se odehrávala v čase po nějakou dobu a má známý začátek a konec události.
- **Epocha** je historické období charakteristické společnými znaky (např. doba kamenná).
- **Zobrazené časové období** je množina událostí, které se udály v daném časovém období. (např. události, které se udály mezi 3.květnem a 17. červnem)
- **Popisek časové osy** je ukazatel, který vizualizuje měřítko zobrazeného časového období.

1.4 Příklady způsobů vizualizace

V této kapitole popíšeme některé z aplikací, které se zabývají vizualizací časových os a to zejména způsob, jakým jsou data vizualizována.

1.4.1 World Anvil

World Anvil [5] je webová aplikace, která umožňuje uživatelům vytvořit si vlastní fiktivní svět. Obsahuje velké množství nástrojů a jedním z nich je nástroj pro vytváření časových os.

Uživatel si může vytvořit události, které se staly v jeho světě. Každá taková událost musí obsahovat název a od kdy do kdy trvala. Všechny uživatelem vytvořené události se zobrazí na ose ve směru odshora dolů v pořadí, ve kterém se udály. Veškeré informace o události jsou vepsány do obdélníku, jehož pozice vůči popisku časové osy a případně označení na popisku časové osy vyjadřují důležitost události. Nejdůležitější události leží přímo na popisku časové osy a veškeré méně důležité události leží vedle a jsou na popisku označeny různými symboly podle důležitosti (čtvereček - velmi důležité, kolečko - triviální). Vzhled události může uživatel ovlivnit výběrem kategorie události. Tento výběr se projeví kruhovou ikonkou v rohu události a barevným proužkem (např. kulturní událost - žlutý proužek, obrázek amfory). Pokročilejší možnosti nastavení umožňují připojit k události odkaz na postavu, organizaci, místo apod., kterých se událost týká. To se projeví v události jako ikonka připojené věci. Editor popisu události umožňuje měnit styl popisu od velikosti nadpisů po vkládání seznamů přes vložení části popisu s varováním, že tato část obsahuje spoiler.

Kromě událostí můžete na osu přidávat epochy. To se projeví větším nadpisem na popisku časové osy a intervalem od kdy do kdy epocha trvala.

Na obrázku 1.2 jsou zobrazeny tři události s různými důležitostmi a různými kategoriemi.

1.4.2 Timeline Maker Pro

Timeline Maker Pro [6] je desktopová aplikace, která umožňuje vytvářet a upravovat neinteraktivní časové osy. Časové osy jsou generovány z tabulky, ve které jsou vyplněny údaje o událostech. Tyto tabulky lze ručně napsat přímo v aplikaci nebo importovat ze souboru ve formátu TXT, CSV nebo MPP. Časové osy lze z aplikace exportovat ve formátu PDF, HTML, TLMZ nebo jako obrázek.

Aplikace umožňuje vytvořit pět druhů časových os. Druhy Flag a Vertical jsou lineární typy, kde Flag využívá horizontální zobrazení (viz. obrázek 1.3) a Vertical vertikální zobrazení. Události jsou ve výchozím stavu vepsány do obdélníků a spojeny šipkou s popiskem časové osy. Aplikace umožňuje nastavit tvar, rámeček a barvu pro každou událost, vzhled šipek spojující události s popiskem časové osy a množství zobrazovaných popisků.

Dalším druhem časových os je Chronology, která je typem mřížka. Tento druh zobrazuje tabulku událostí v chronologickém pořadí. Čtvrtým druhem je lineární vizualizace jménem Bar.



■ Obrázek 1.2 Část časové osy ve World Anvil

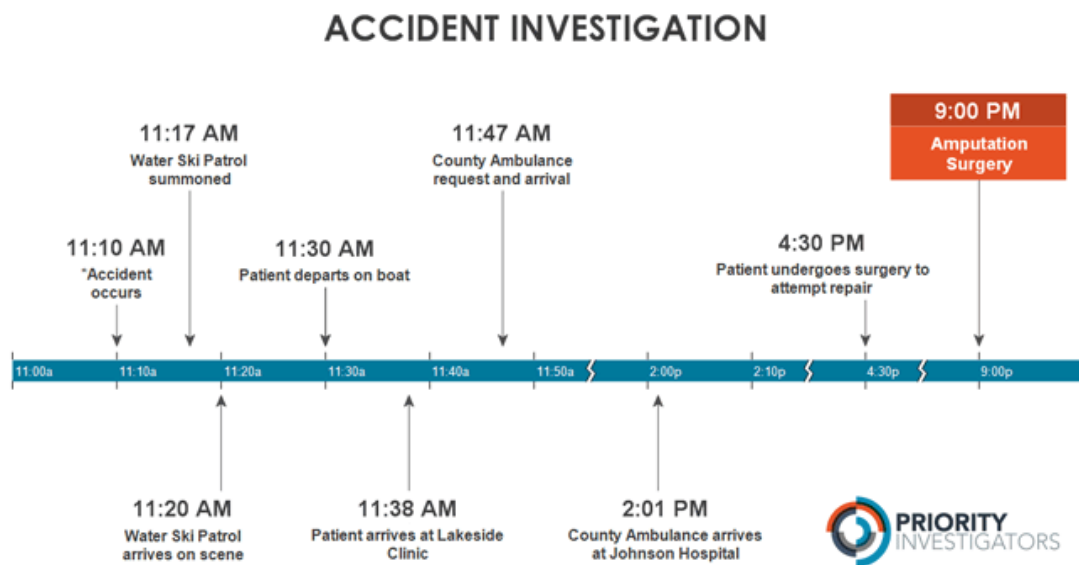
V této vizualizaci jsou na horním a spodním okraji zobrazeny popisky časové osy a mezi nimi jsou v obdélnících zobrazeny události. Délka obdélníku značí, jak dlouho událost trvala a jeho pozice vůči popisku časové osy značí kdy událost začala a kdy skončila. Události jsou vykreslovány do jednoho řádku. V případě, že by se události překrývaly, jsou události vykresleny pod sebe. Jednotlivým událostem a popiskům časové osy lze měnit tvar, barvu, rámeček a stín.

Posledním druhem je Project Gantt, který kombinuje druhy Chronology a Bar. V levé části vizualizace jsou vypsány v tabulce chronologicky za sebou všechny události a v pravé části jsou odshora dolů vykresleny události obdobným způsobem jako v Bar, ale každá událost má svůj řádek.

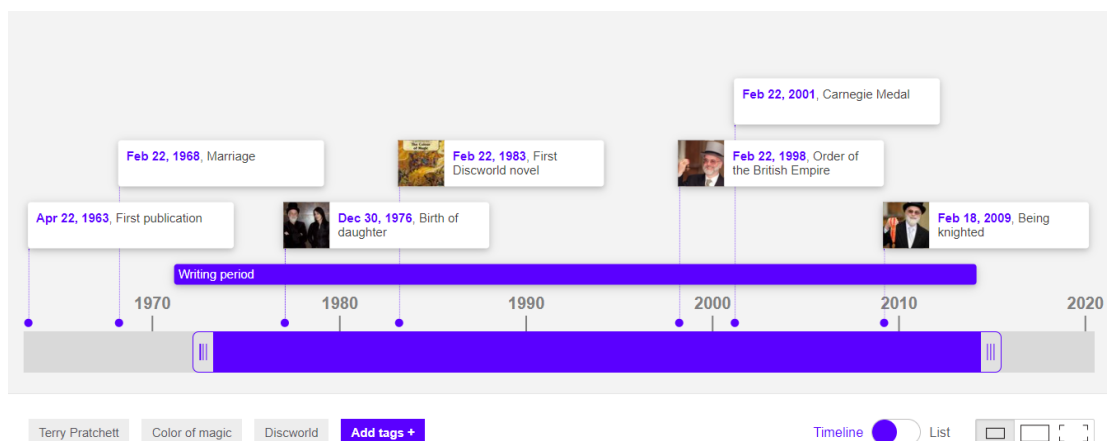
1.4.3 Timetoast

Timetoast [7] je webová aplikace pro tvorbu interaktivních časových os. Aplikace umožňuje přepínat mezi dvěma způsoby vizualizace. V režimu Timeline je použita horizontální časová osa (viz obrázek 1.4). Každá událost je vykreslena jako obdélník s přímkou kolmou na popisek časové osy, kde je zakončena puntíkem. Tento obdélník obsahuje obrázek, název události a datum, kdy se událost odehrála. Po kliknutí na událost je zobrazen podrobnější popis této události. V případě, že by se události překrývaly, jsou události vykresleny nad sebe. Vizualizace umí vykreslit také epochy a to jako proužek s délkou odpovídající délce epochy na popisku časové osy. Pro úpravu zobrazeného časového úseku lze použít fialový pruh pod časovou osou. Tento pruh lze posouvat a měnit jeho délku, čímž se mění zobrazený časový úsek a jeho délka.

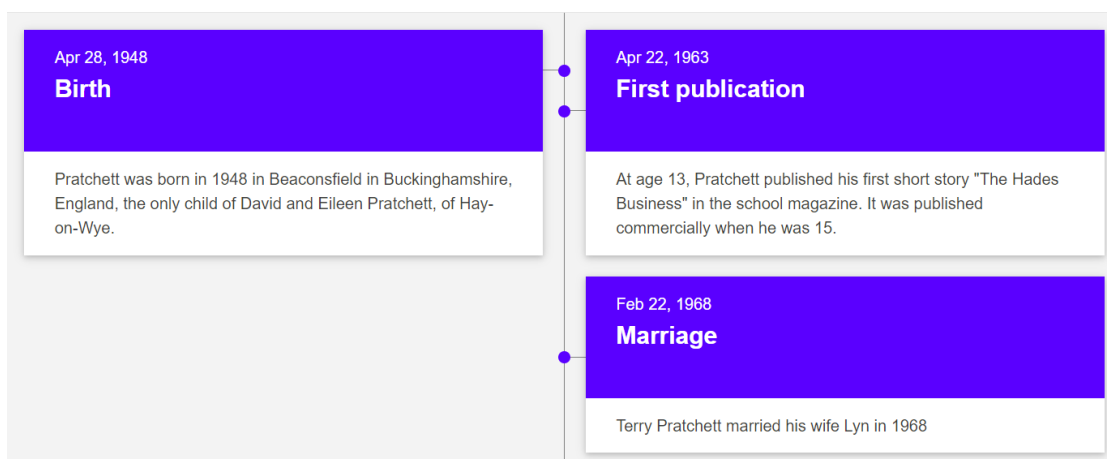
V režimu zobrazení List (viz obrázek 1.5) je využita vertikální časová osa. Vizualizace je rozdělena uprostřed vertikální přímkou. Události i epochy jsou vykresleny jako obdélníky po stranách této přímky v chronologickém pořadí. Přímka nemá žádný popisek a události jsou vykreslovány vždy ve stejné vzdálenosti od sebe, tedy umístění události vůči přímce nemá spojitost s tím, kdy se udála.



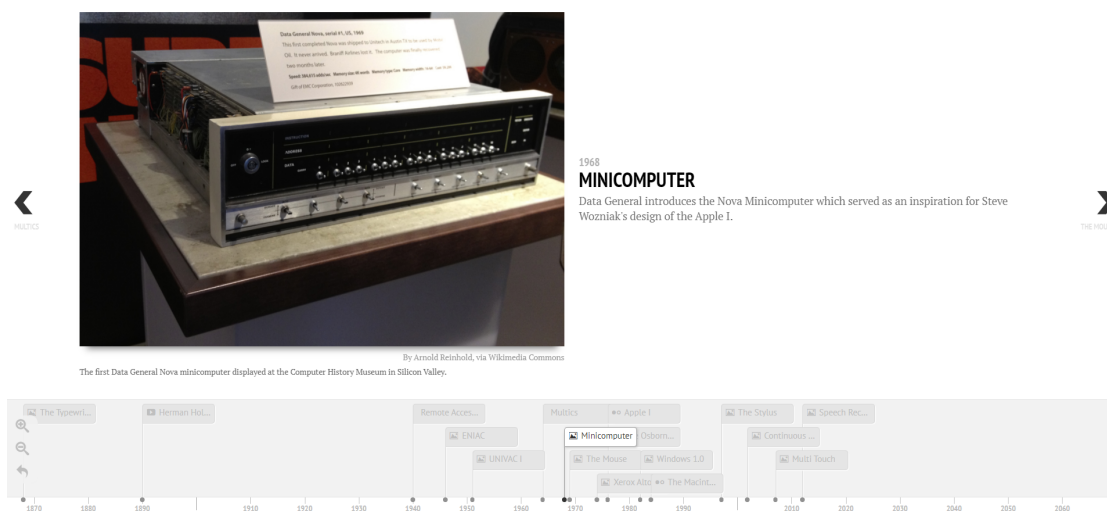
■ **Obrázek 1.3** Časová osa v Timeline Maker Pro, zobrazení Flag



■ **Obrázek 1.4** Část časové osy v Timetoast v režimu Timeline



■ Obrázek 1.5 Část časové osy v Timetoast režimu List



■ Obrázek 1.6 Časová osa ve TimelineJS

1.4.4 TimelineJS

TimelineJS [8] je open-source nástroj pro generování časových os a jejich vkládání do webových aplikací. Pro vygenerování časové osy lze použít Tabulky Google nebo JSON soubor s vhodně vyplněnými údaji. Při generování lze nastavit použitá písma, přednastavení zobrazeného časového období a velikost vizualizace. Pro pokročilejší nastavení vzhledu je potřebná znalost programovacího jazyka CSS.

Vizualizaci (viz obrázek 1.6) můžeme rozdělit na dvě části. V horní části je zobrazena aktuálně vybraná událost, kterou tvoří nadpis, popis, datum a případně obrázek nebo jiný typ média (video, audio, google maps...). Pro zobrazení přehledu událostí slouží spodní část vizualizace, kterou tvoří popisek časové osy a nad ním prostor pro události. Každá událost je zobrazena jako obdélník s názvem události s přímkou směřující na popisek časové osy do bodu, ve kterém se událost odehrála. V případě, že by se obdélníky překrýval, se vykreslí události nad sebe. Zobrazené časové období lze posunout, zvětšit nebo zmenšit. Vybraná událost lze změnit kliknutím na jinou událost nebo použitím šipek pro vybraní události, která je chronologicky nejbližší vybrané události.

1.4.5 Tiki-Toki

Tiki-Toki [9] je webová aplikace pro vytváření interaktivních časových os. Zobrazení má dva základní režimy - 2D a 3D. Ve 2D režimu (viz obrázek 1.7) jsou události zobrazeny jako obdélníky se zobáčkem ukazujícím kolmo na popisek časové osy v dolní části vizualizace, do místa, ve kterém se událost začala odehrávat. V obdélníku může být vepsán název, popis, od kdy do kdy se událost odehrála, obrázek, video a nebo zvukový soubor. Každá událost dále obsahuje vždy tlačítko. Po kliknutí na něj se zobrazí podrobnější popis události. Události lze také rozdělit do kategorií, což se projeví proužkem s barvou přiřazenou ke kategorii. Kromě událostí jsou na časové ose zobrazené epochy, které se projevují změnou pozadí v místě, kde na časové ose probíhají. Ve 3D režimu (viz obrázek 1.8) je popisek časové osy zobrazen jako dlouhá plošina směřující od uživatele do dálky. Události jsou vykresleny za sebou jako plovoucí objekty nad tímto popiskem. Čím dále směrem od uživatele je umístěna událost, tím dříve v historii se událost udála.

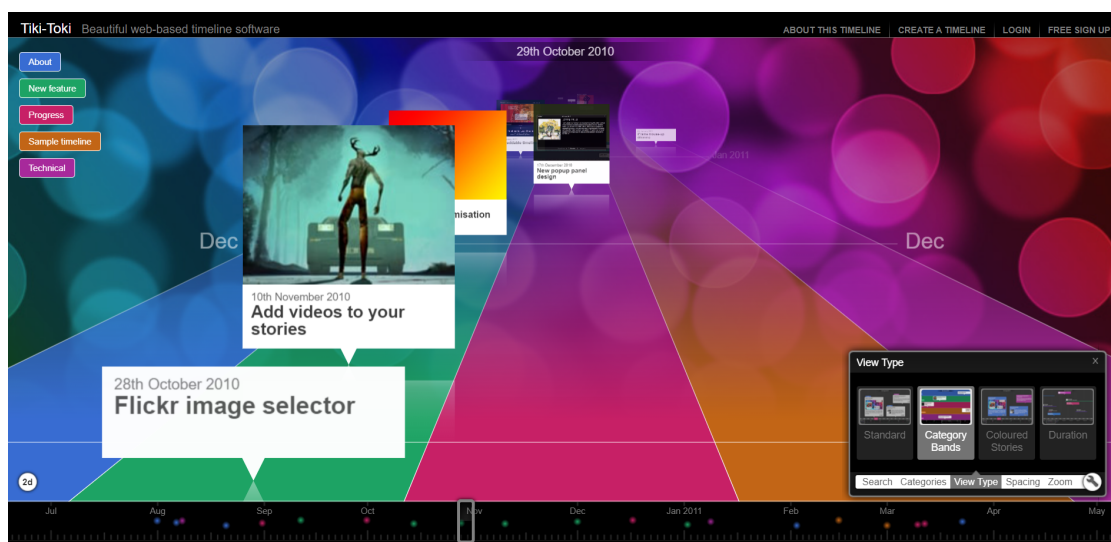
Oba dva režimy zobrazení mají další 4 způsoby zobrazení, které upravují vzhled vizualizace. Základním zobrazením je Standard, který byl popsán výše a zobrazuje ho obrázek 1.3. Dalším způsobem zobrazení je Coloured Stories, které mění barvu obdélníků, ve kterých jsou zobrazeny události, na barvu kategorie, do které událost patří. U událostí poté není vykreslen proužek určující, do které kategorie událost patří, protože je to zřejmé z barvy obdélníku. Třetím způsobem zobrazení je Duration, který trvající události vykresluje jako úzký pruh, který začíná a končí v bodech, které odpovídají začátku a konci události na popisku časové osy. Posledním způsobem zobrazení je Category Bands (viz obrázek 1.8), které kombinuje předchozí dva způsoby a k tomu rozděluje popisek časové osy do barevných pruhů, které představují jednotlivé kategorie událostí. V každém pruhu jsou poté zobrazeny pouze události, které jsou v této kategorii.

Ve vizualizaci je ve spodní části obrazovky vykreslena navigační časová osa, která zobrazuje delší časové období a slouží pro posunutí zobrazeného období hlavní časové osy. Na navigační časové ose je vykreslen šedý rámeček, který značí, který časový úsek zobrazuje hlavní osa. Dále jsou na této ose vykresleny barevné tečky představující jednotlivé události a barevné pruhy představující epochy.

Pro posunutí zobrazeného časového úseku na hlavní ose lze také použít šipky na klávesnici a myš. Délku zobrazeného časového období lze změnit v nastavení v sekci Zoom. Množství zobrazených událostí lze upravit v nastavení pomocí zobrazení pouze vybraných kategorií nebo pomocí vyhledání událostí, které obsahují hledaný výraz.



■ Obrázek 1.7 Časová osa v aplikaci Tiki Toki ve 2D režimu



■ Obrázek 1.8 Časová osa v aplikaci Tiki Toki ve 3D režimu

Analýza podobných aplikací

V této kapitole zhodnotím jednotlivé vizualizace z aplikací popsaných v předchozí kapitole. Hlavními kritérii mého hodnocení bude vzhled vizualizace, obtížnost orientace v zobrazených událostech, možnosti filtrace událostí a interakce s časovou osou.

2.1 World Anvil

World Anvil [5] vyniká především svým příjemným vzhledem a širokými možnostmi při tvorbě časové osy. Možností interakce s hotovou časovou osou ale není moc, což při velkém množství událostí ubírá ose na přehlednosti. Osa je spíše vhodná, pokud si chce uživatel něco zajímavého přečíst, ale pro vyhledávání konkrétních událostí je docela nešikovná. K nalezení konkrétní události může pomoci rozdělení událostí do kategorií, které jsou jasně barevně vyznačeny, ale každá událost zabírá hodně prostoru na obrazovce a pohyb po ose je pomalý. Pokud si chce uživatel přečíst něco o poslední události, nezbývá mu než srolovat obrazovku až dolů.

Klady:

- + Příjemný vzhled
- + Velký výběr kategorií událostí
- + Každá kategorie má svou ikonu a barvu
- + Pěkné písmo
- + Možnost připojit postavu, organizaci apod.
- + U událostí lze nastavit jejich důležitost
- + Tlačítko pro návrat na začátek časové osy

Zápory:

- Žádná možnost vyhledávání nebo filtrování událostí na časové ose
- Pozice událostí nevyjadřuje, kdy se událost udála
- Žádná možnost interakce s časovou osou
- Posun rozsahu zobrazených událostí je možný pouze posouváním na stránce, což je při velkém množství událostí zdlouhavé

2.2 TimelineJS

TimelineJS [8] má velmi jednoduchý vzhled, který se nedá upravit jinak než pomocí CSS souboru, což vyžaduje znalost programovacího jazyka CSS. Pohyb po ose je podstatně snazší než u World Anvil [5] a to především díky malému rozměru událostí a možnosti měnit rozsah zobrazeného časového úseku. Při velkém množství dat se události překrývají a není snadné se v nich orientovat. Všechny události jsou na ose zobrazeny stejným způsobem a nalezení konkrétní události není vůbec snadné.

Klady:

- + Na časové ose se lze snadno posouvat, obzvláště po sousedních událostech
- + Snadná změna rozsahu zobrazených událostí
- + Tlačítko pro návrat na začátek časové osy

Zápory:

- Nemá moc pěkný vzhled
- Žádná možnost vyhledávání nebo filtrování událostí na časové ose
- Při velké hustotě událostí se události překrývají

2.3 Timeline Maker Pro

Timeline Maker Pro [6] slouží spíše pro vytvoření obrázku časové osy do prezentace než pro vytvoření interaktivní osy, na které by se daly vyhledávat informace a nelze ho moc srovnávat s ostatními popisovanými aplikacemi. Vzhled os lze velmi snadno nastavit pro každou událost zvlášť a to i prvky, které ostatní aplikace neumožňují, a nabízí zajímavé způsoby vizualizace dat. Při malém množství dat jsou časové osy velmi přehledné. Tím ale výhody končí. Vzhledem k tomu, že jsou osy neinteraktivní, nejsou vhodné pro velké množství dat a brzy se v nich špatně orientuje.

Klady:

- + Vzhled lze upravit
- + Rozdělení událostí do kategorií
- + Více způsobů, jak zobrazit osy
- + Umožňuje zobrazit události, které se odehrávaly v čase

Zápory:

- Úplně neinteraktivní, časová osa je pouze obrázek
- Vhodné pouze pro malé množství událostí

2.4 Tiki-Toki

Tiki-Toki [9] je podle mě velmi zajímavá aplikace díky unikátním způsobům vizualizace. Za zajímavý považuji 3D režim, který ukazuje intuitivně, jak dlouhá doba uběhla mezi událostmi. Pro velké množství událostí není ale přehledný, protože se události navzájem překrývají. Dalším zajímavým režimem je Category bands, při kterém se časová osa rozdělí na pruhy a v každém pruhy jsou události pouze jedné kategorie. To může při vhodně zvoleném množství kategorií udělat osu přehlednější.

Pohyb po časové ose je usnadněn pomocí náhledu časové osy pro velké časové období. V tomto náhledu jsou zobrazeny všechny události jako puntík v barvě, která odpovídá kategorii události. Možnost filtrovat události pomocí vyhledávače a pomocí výběru kategorie událostí umožňuje najít na ose konkrétní hledané události.

Klady:

- + Zajímavé způsoby vizualizace
- + Umožňuje zobrazit trvající události
- + Rozdělení událostí do kategorií
- + Pohyb po ose je snadný díky náhledu časové osy
- + Nastavitelný rozsah zobrazeného časového období
- + Filtrace událostí pomocí vyhledávání
- + Filtrace událostí pomocí kategorií
- + Hezké zobrazení epoch

Zápory:

- Při velkém množství dat se události překrývají

2.5 Timetoast

Timetoast [7] vyniká velmi intuitivním posuvníkem, kterým lze velmi snadno a rychle měnit rozsah zobrazeného časového období, což velmi zjednodušuje orientaci ve větším množství událostí a pohyb po ose. I přesto hledání konkrétních událostí není snadné, protože aplikace neumožňuje žádné možnosti vyhledávání a filtrování událostí. Vzhled vizualizace ničím nevyniká.

Klady:

- + Snadná a intuitivní změna zobrazeného časového období v režimu Timeline
- + Události se nepřekrývají

Zápory:

- Žádná možnost vyhledávání nebo filtrování událostí na časové ose
- Vzhled vizualizace není ničím zajímavý

2.6 Závěr

Ze všech aplikací, které byly zhodnoceny v této kapitole mi přišly nejzajímavější ikonky pro každou z kategorií událostí v aplikaci World Anvil [5], náhled časové osy v aplikaci Tiki-toki [9] a interaktivní posuvník v aplikaci Timetoast [7].

Rešerše technologií

V této kapitole popíší technologie, které je možné použít pro webovou vizualizaci a všechny technologie pro to potřebné.

3.1 HTML, CSS a JS

Dle [10] HTML, CSS a Javascript jsou třemi základními pilíři tvorby webových stránek.

Podle Mozilla [11] je HTML značkovací jazyk, který se používá k vytváření struktury obsahu webové stránky. HTML obsahuje sérii elementů, které slouží k uzavření či oddělení částí obsahu s cílem odlišit vzhled a chování těchto částí. Párové značky umožňují změnit velikost písma, změnit jeho tloušťku, vytvořit odkaz apod. Vkládáním značek do sebe lze vytvořit složitější konstrukce jako jsou například číslované seznamy. HTML elementům lze nastavit třídu, id a další atributy.

Mozilla [11] říká o CSS, že jsou to kaskádové styly. CSS se používá k vytváření stylů vybraných elementů HTML. Jeho zápis umožňuje vybrat HTML elementy podle typu HTML elementu, id, třídy nebo jiného atributu. Díky CSS lze změnit barvu textu, jeho font a velikost, barvu pozadí, odsazení HTML elementů od sebe a mnoho dalšího.

Javascript je podle Mozilla [11] skriptovací jazyk, který přidává interaktivitu do webových stránek. Umožňuje dynamicky měnit obsah, kontrolovat multimédia, animovat obrázky a mnoho dalšího. Javascript umožňuje používat velké množství knihoven, díky kterým lze například manipulovat s HTML elementy, pracovat se zvukovými a video soubory a v neposlední řadě také vytvářet 2D a 3D grafiku. Pro to lze buď použít Canvas API nebo WebGL API.

3.2 Canvas, WebGL a SVG

Canvas API je podle Mozilla [11] API, které umožňuje vykreslovat grafiku pomocí Javascriptu do HTML elementu Canvas. Toto API je zaměřené především na 2D grafiku a umožňuje vytvářet animace, vizualizovat data, pracovat s fotografiemi, vykreslovat herní grafiku a zpracovávat videa.

WebGL (Web Graphics Library) je výkonná knihovna pro vykreslování 2D a 3D grafiky pomocí Javascriptu a s možností použití HTML elementu Canvas. WebGL dokáže využít hardwarovou grafickou akceleraci poskytovanou zařízením uživatele.

SVG (Scalable Vector Graphics) je podle Mozilla [11] značkovací jazyk pro tvorbu vektorové grafiky. SVG obrázky a jejich chování jsou zapsány ve formátu XML. Díky tomu že SVG vykresluje vektorovou grafiku, jsou SVG obrázky vždy ostré, ať jsou vykresleny v jakékoli velikosti.

Všechny tři uvedené způsoby vykreslování jsou podle Yworks [12] nejvhodnější k použití ve specifických případech. SVG je ideální řešení pokud chceme vykreslit pouze pár grafických prvků a chceme, aby byly ostré při jakémkoli zvětšení. Canvas je vhodný pro větší množství prvků a

pro jejich animace, ale počet grafických prvků je stále omezený. WebGL je ideální pro zobrazení hodně velkého množství prvků včetně jejich animace a 3D vizualizací. Vzhledem k tomu, že ve vizualizaci budou moci být zobrazené desítky událostí, bude podle mě vhodné použít spíše Canvas nebo WebGL.

3.3 Knihovny pro 2D grafiku

V této části stručně popíši některé z knihoven pro práci s 2D grafikou, které by mohly být vhodné pro vizualizaci časových os a v závěru vyberu jednu, pomocí které bude vizualizace implementována.

PixiJS

Podle dokumentace [13] je PixiJS knihovna pro vykreslování 2D grafiky, která využívá WebGL a Canvas. Knihovna je velmi rychlá a umožňuje vykreslit bohatou interaktivní grafiku. Knihovna poskytuje různé režimy prolnutí, tónování, pokročilé možnosti vykreslování textu a tvorbu a použití WebGL filtrů.

Phaser

Phaser, jak říká dokumentace [14] je také grafická knihovna používající WebGL. Oproti PixiJS je zaměřená hlavně na tvorbu her a obsahuje rozhraní pro práci s fyzikou, zvukem a částicemi a poskytuje více možností jak pracovat se vstupem od uživatele.

D3.js

D3.js je podle stránek D3.js [15] knihovna pro interaktivní vizualizaci dat pomocí SVG, Canvas, HTML a CSS. Knihovna umožňuje snadno nakreslit různé typy grafů, ale i vytvořit vlastní interaktivní a animované vizualizace dat.

Two.js

Two.js je podle two.js dokumentace [16] knihovna pro vykreslování, která umožňuje použít SVG, Canvas i WebGL. Použitím této knihovny lze snadno vykreslovat jednoduché tvary a animovat je.

3.4 Závěr

Ve vizualizaci časových os budou vykresleny desítky událostí, které budou mít animaci při změně zobrazeného rozsahu, což umožňují všechny popsané knihovny. Knihovna Phaser navíc přidává i práci s fyzikou, která pro vizualizaci nebude potřeba a při vykreslování vystačím s jednodušší knihovnou. Two.js nemá moc obsáhlou dokumentaci o vstupu od uživatele a proto bych ráda použila spíše některou ze zbylých knihoven. D3.js se mi zdá pro vykreslování časových os velmi vhodná, ale vzhledem k tomu, že pomocí PixiJS byl vytvořen prototyp aplikace a knihovna se osvědčila, vyberu si pro vytvoření vizualizace tu. Pro ovládací prvky vizualizace použiji HTML a CSS.

Historická data Asterionu

V této kapitole popíšeme získaná historická data ze světa Asterion a navrhnu rozdělení těchto dat do entit. Pro toto rozdělení následně popíšeme konkrétní metodiku získání dat pro entity z dat získaných od zadavatele.

4.1 Prostředí Asterionu

Svět Asterion je na stejnojmenné planetě Asterion. Na planetě leží dva světadíly, Lendor (viz obrázek 4.1) na severu a Tara na jihu (viz obrázek 4.2). Na obou těchto kontinentech i v okolních oceánech se rozkládá spousta zemí, které jsou domovem velkého množství ras např. lidí, elfů, trpaslíků, skřítků, heverenů a dalších. V dějinách Asterionu se odehrálo velké množství událostí od válek, přes narození panovníků a jejich skonání, po rozpady království.

4.1.1 Kalendářní systém

Asterion má svůj vlastní kalendářní systém, který je složitější než gregoriánský kalendářní systém. Jeden rok Asterionu trvá dvanáct měsíců¹ a každý z těchto měsíců má přesně 30 dní. Rok má 370 dní, z nichž deset nepatří do žádného měsíce. Tyto dny jsou svátky², které se odehrávají vždy mezi měsíci a nepatří do žádného z nich. Svátky nejsou součástí ani týdne, který na Asterionu trvá deset dní³.

Z důvodu větší komplikovanosti práce s daty, budou ve webové aplikaci data reprezentována jako vzdálenost od 1. Chladna roku 0 udaná ve dnech. Například 1. Novorost roku 0 bude zaznamenán jako den 61 (vzdálenost dva měsíce po 30 dnech a jeden svátek).

4.2 Rozdělení dat na entity

Pro práci s historickými daty Asterionu rozdělím tato data do entit, z nichž každá bude popisovat pouze část z historických dat. Jedním z typů entit bude Událost, která je pro zobrazování historických událostí na časové osy nejdůležitější.

¹Chladen, Kliden, Novorost, Rozkvet, Zelenec, Ploden, Žluten, Úmor, Traven, Ovocen, Větrnec a Dešten

²Den půstu, Den očištění, Svátek letních duchů (trvá dva dny), Svátek mladých srdcí, Den hojnosti, Den zákonů, Svátek zimních duchů (trvá dva dny) a Svátek proroků

³den dubu, den koně, den buku, den vlka, den vrby, den skřivana, den berana, den orla, den břízy, den jasanu



■ Obrázek 4.1 Mapa Lendoru



■ Obrázek 4.2 Mapa Tary

4.2.1 Události

Entita typu Událost bude představovat jednu událost v historii Asterionu. Každá entita tohoto typu bude mít název, který bude výstižně a krátce popisovat, co se během této události událo, datum, kdy událost nastala a popisek, který bude podrobně popisovat průběh události.

4.2.2 Tagy

Každá událost obsahuje nějaká klíčová slova, která jsou pro význam události důležitá. Těmto klíčovým slovům budu nadále říkat tagy. Entita Tag bude zahrnovat název tagu, který bude krátce a výstižně pojmenovávat podstatu tagu a kategorii, která bude jednoznačně určovat o jaký typ tagu se jedná. Např. pro událost „Karel IV. nechal postavit hrad Karlštejn“ můžeme vytvořit dva tagy. První bude mít název „Karel IV.“ a kategorii „osoba“ nebo „panovník“ a druhý tag může mít název „Karlštejn“ a kategorii „místo“ nebo „hrad“ dle použité granularity.

4.3 Metodika tvoření entit

Svět Asterion má všechny významné události od roku 846 do roku 873 podrobně popsány v kronice Asterionu, kterou jsem získala od zadavatele. V této kronice každá kapitola představuje jeden rok a dále je v některých případech členěna podle míst, ve kterých se události odehrály. Všechny události jsou vypsány v chronologickém pořadí, dělené do odstavců. Některé odstavce popisují i více událostí, které se odehrávaly těsně po sobě, nebo spolu úzce souvisí jiným způsobem.

Rozdělení kroniky do odstavců umožňuje snadno vyseparovat z textu popis události. Poté je nutné určit, co je hlavním významem tohoto popisu a shrnout ho do krátkého názvu. Díky dělení kroniky do kapitol podle roků, je rok, kdy se událost stala, vždy známý. V případě, že u události není zřejmé, v jaký den nastala, je nutné vybrat datum náhodně a zachovat při tom pořadí událostí, v jakém jsou zapsány v kronice.

852

Černá vdova se vrací do Albirea, kde upevňuje a rozšiřuje svou síť. Její střet se Zvonisladem, bývalým vůdcem Nočního letu a nyní agentem Almendorské tajné služby, pověřený eliminací Pavučiny, se nevyhnutelně blíží.

Oproti veškeré snaze lesních elfů se jejich domovy stále šíří neznámá hrůza. Přes jedinou noc bylo v Platanovém háji spáleno nevídaným mrazem několik stříbrných platanů. Na pomezí Červeného lesa operují malé skřeti záškodnické skupiny, vždy udeří a zmizí kdesi v Podzemní říši.

Kirbeg rychle opravuje svou flotilu. V přístavu na Ostrově albatrosů bylo spatřeno několik plavenských lodí.

■ Obrázek 4.3 Ukázka z kroniky Asterionu

Po získání celé události, je nutné určit tagy. Většinou mezi tagy budou patřit všechna vlastní jména vyskytující se v popisu události a další slova nebo slovní spojení, která jsou pro událost klíčová. Tag je poté nutné zařadit do kategorie.

Na obrázku 4.3 je ukázka z kroniky Asterionu. Tato část by podle uvedené metodiky tvořila tři události a jejich popis by mohl být následující:

První událost

- **Název události:** Černá vdova se vrací do Albirea
- **Popis:** Černá vdova se vrací do Albirea, kde upevňuje a rozšiřuje svou síť. Její střet se Zvonisladem, bývalým vůdcem Nočního letu a nyní agentem Almendorské tajné služby, pověřený eliminací Pavučiny, se nevyhnutelně blíží.
- **Datum:** 3. 2. 852
- **Tagy:** Černá vdova (osoba), Albireo (země), Zvonislad (osoba), Pavučina (organizace), Almendorská tajná služba (organizace)

Druhá událost

- **Název události:** Lesní elfové jsou napadání skřety
- **Popis:** Oproti veškeré snaze lesních elfů se jejich domovy stále šíří neznámá hrůza. Přes jedinou noc bylo v Platanovém háji spáleno nevídaným mrazem několik stříbrných platanů. Na pomezí Červeného lesa operují malé skřeti záškodnické skupiny, vždy udeří a zmizí kdesi v Podzemní říši.
- **Datum:** 17. 6. 852
- **Tagy:** lesní elfové (rasa), Platanový háj (les), Červený les (les), skřeti (rasa), Podzemní říše (země)

Třetí událost

- **Název:** Kirbeg opravuje svou flotilu
- **Popis:** Kirbeg rychle opravuje svou flotilu. V přístavu na Ostrově albatrosů bylo spatřeno několik plavenských lodí.
- **Datum:** 22. 8. 852
- **Tagy:** Kirbeg (osoba), Ostrov albatrosů (ostrov), Plavena (země)

Návrh webové aplikace

V této kapitole budu popisovat návrh vizualizace a uživatelského rozhraní. Webová aplikace bude obsahovat dvě webové stránky. Stránka Asterion Timelines bude pro běžné uživatele a bude obsahovat vizualizaci časových os. Tato stránka bude rozdělena do tří částí. V první části bude zobrazena hlavička stránky, v druhé části bude rozhraní pro ovládání vizualizace a v poslední části samotná vizualizace.

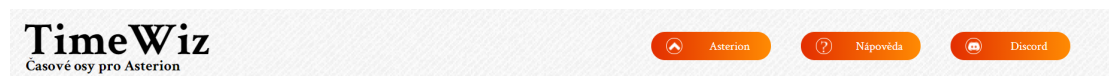
Druhá ze stránek bude sloužit administrátorům aplikace pro úpravu a přidávání nových událostí, tagů, kategorií tagů a pod.

5.1 Návrh stránky Asterion

V této sekci se budu zabývat návrhem vizualizace událostí na časové osy. To zahrnuje návrh vzhledu časové osy, způsobu zobrazení událostí a vzhledu událostí.

5.1.1 Hlavička stránky

Hlavičku webové stránky bude tvořit název stránky a odkazy na webové stránky, které jsou významné pro komunitu hráčů RPG her ve světě Asterion. Návrh hlavičky je vyobrazen na obrázku 5.1.



■ Obrázek 5.1 Návrh hlavičky webové aplikace

5.1.2 Zobrazení časové osy

Všechny aplikace, které byly popsány v předchozích kapitolách, využívají k vizualizaci jednu časovou osu a všechny události se zobrazují na ní. Toto zobrazení je obvykle použito pro události s jedním společným tématem (např. biografie jedné osoby, historie firmy). Vzhledem ke komplikovanosti a rozsáhlosti historie Asterionu není možné zobrazit všechny události na jednu časovou osu najednou, protože orientace v takovéto vizualizaci by byla velmi náročná. Možným řešením je rozřadit události do kategorií nebo každou událost označit klíčovými slovy a frázemi. Poté můžeme nechat uživatele určit, co bude na časové ose zobrazeno.

Při řešení tohoto projektu bych ráda využila hlavně druhý ze způsobů. Ke každé události budou přidány štítky (dále tagy), tedy slova nebo slovní spojení, která jsou pro událost významná (např. událost „Karel IV. nechal postavit hrad Karlštejn“ by mohla mít tagy „Karel IV.“, „Karlštejn“ a „hrad“). Uživatel si bude moci vybrat mezi tagy jeden a poté budou na časové ose vykresleny pouze události označené tímto tagem.

Vzhledem k tomu, že cíloví uživatelé budou i spisovatelé, kteří píšou knihy z prostředí Asterionu, může pro ně být podstatné najít spojitosti mezi více tagy (např. zda-li se dvě postavy mohly setkat). A proto bych ve vizualizaci umožnila zobrazit více časových os najednou pod sebou, kde by na každé ose byly zobrazeny události označené uživatelem zvoleným tagem. Události, které budou mít časové osy společné, budou pro vyšší přehlednost spojeny přímkou.

5.1.3 Vzhled časové osy

Vzhledem k zobrazení více časových os najednou, bude muset být zobrazení časové osy minimalistické. Časovou osu bude tvořit přímka, na které budou zobrazeny události, a ikona s popiskem, která bude sloužit pro usnadnění orientace v přímkách. Ikona u časové osy bude určena výběrem tagu (např. modrý erb s jednorozcem pro Danérii jako je to na obrázku 5.2). Každý tag může mít vlastní ikonku a v případě, že ji mít nebude, bude určena pomocí kategorie, do které tag patří. Uživatel si bude moci, pro lepší orientaci, nastavit barvu zobrazené přímky.



■ Obrázek 5.2 Návrh časové osy

5.1.4 Událost

Události budou vykresleny jako kruhové ikonky na přímkě s názvem události pod touto ikonkou (viz obrázek 5.3). Každá událost bude mít přiřazenou ikonku podle tématiky této události, např. „Narození Bogaje“ může mít ikonku novorozence. Podrobnější popis události bude dostupný při najetí kurzorem na ikonku události.



■ Obrázek 5.3 Návrh událostí

5.1.5 Popisek časové osy

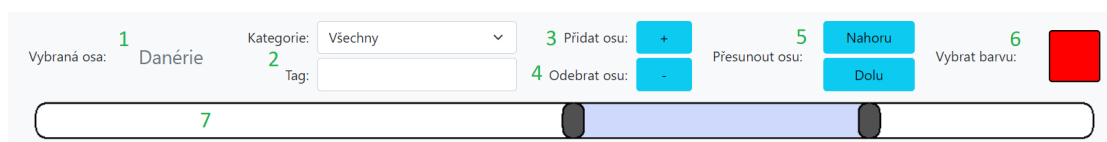
Vizualizace časových os bude mít na horní a spodní straně zobrazený popisek časových os, který bude, pomocí pruhů vedoucích přes vizualizaci, rozdělovat osy na stejně dlouhé úseky. U každého pruhu bude v horní a spodní části uvedeno jemu příslušné datum. Na obrázku 5.4 je zobrazena část vizualizace s popiskem v horní části.



■ **Obrázek 5.4** Návrh popisku časové osy

5.1.6 Rozhraní pro ovládání vizualizace

V této sekci popíšu potřebné rozhraní pro ovládání vizualizace a úpravu aktuálně vybrané osy. Úprava aktuálně vybrané časové osy zahrnuje změnu vybraného tagu, změnu barvy časové osy a výměnu časové osy s osou pod ní a nad ní. Nastavení celé vizualizace bude umožňovat přidávání a odebrání časových os a změnu zobrazeného období na všech časových osách. Všechny ovládací prvky potřebné k těmto činnostem a jejich popisky budou obsaženy v kontrolním panelu, který se bude nacházet mezi hlavičkou stránky a vizualizací časových os.



■ **Obrázek 5.5** Návrh ovládacích prvků v kontrolním panelu

5.1.6.1 Vybraný tag a jeho změna

Název aktuálně vybraného tagu bude zobrazen jako nápis v levé části panelu jako je tomu na obrázku návrhu kontrolního panelu 5.5 pod číslem 1.

Změna tagu bude umožněna dvěma ovládacími prvky a to vyhledávačem a menu, které jsou na obrázku 5.5 pod číslem 2. Uživatel si bude moci vybrat, pomocí vyhledávače s našeptáváním, libovolný existující tag. Pro usnadnění vyhledávání ve vyhledávači bude sloužit menu, ve kterém si uživatel bude moci zvolit kategorii hledaného tagu a poté mu bude vyhledávač našeptávat pouze tagy v této kategorii.

5.1.6.2 Přidávání a odebrání časových os

Přidávání nových časových os bude umožněno tlačítkem, které je na obrázku 5.5 pod číslem 3. Kliknutím na tlačítko se přidá na konec vizualizace nová prázdná časová osa. Prázdná časová osa nemá zvolený žádný tag a má nastavenou výchozí barvu a výchozí ikonu.

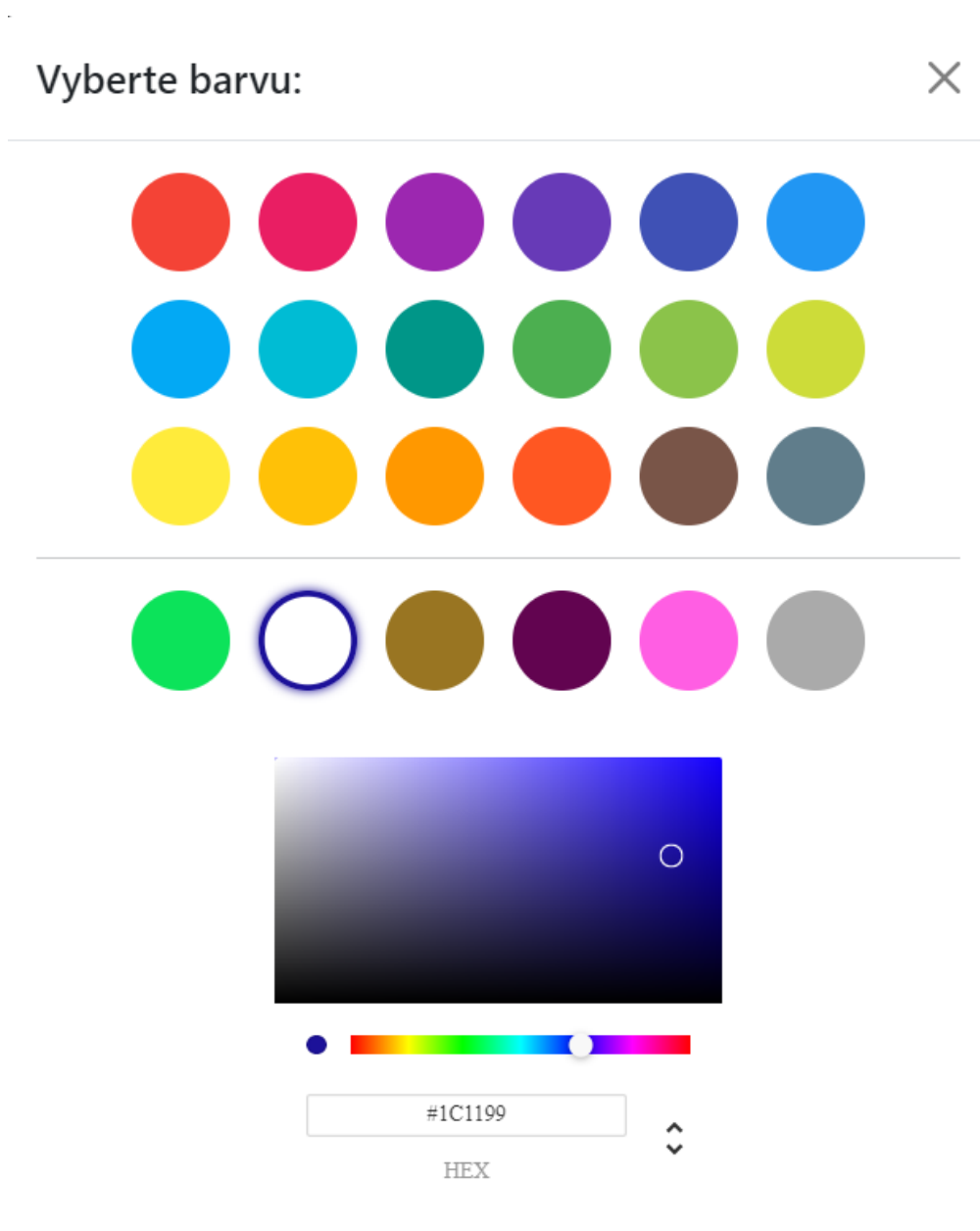
Tlačítko s číslem 4 na obrázku 5.5 bude umožňovat uživateli odebrat aktuálně vybranou osu. Vybraná časová osa se změní na osu, která se nacházela nad smazanou časovou osou, pokud taková osa existovala. Pokud taková osa neexistovala, tak se změní na časovou osu, která se nacházela pod smazanou časovou osou. V případě, že časová osa byla ve vizualizaci poslední, bude osa nahrazena novou prázdnou časovou osou.

5.1.6.3 Posun časové osy

Tlačítka u čísla 5 na obrázku 5.5 budou umožňovat uživateli změnit pořadí časových os. Při kliknutí na tlačítka se aktuálně vybraná časová osa prohodí s osou nad sebou nebo pod sebou, dle použitého tlačítka.

5.1.6.4 Výběr barvy časové osy

Uživatel si bude moci pomocí tlačítka s číslem 6 na obrázku 5.5 zobrazit okno pro výběr barvy přímky aktuálně vybrané časové osy. To bude moci učinit buď výběrem barvy z předem připravené sady barev, nebo si bude moci vytvořit vlastní sadu barev. Na obrázku 5.6 je v horní části zobrazena sada barev pro výběr barvy z předem připravených barev a ve spodní části je sada barev, které lze změnit pomocí výběru barvy pod touto sadou.



■ **Obrázek 5.6** Návrh okna pro výběr barvy

5.1.6.5 Nastavení zobrazeného období

Jednou z možností, jak změnit aktuálně zobrazené časové období ve vizualizaci časových os, je vytvořit formulář, ve kterém si uživatel sám vybere časový interval, který si chce zobrazit. Toto nastavování má nevýhodu v tom, že uživatel musí vědět, v jakém časovém období se odehrály události pro jeho zvolené tagy a při volbě špatného časového období neuvidí žádné události. Výhodou tohoto řešení je snadná implementace, která vyžaduje pouze jednoduchý formulář. Tato metoda byla použita v předchozích verzích prototypu.

Druhou možností je použít interaktivní posuvník, podobný jako využívá aplikace Timetoast [7]. Tento posuvník má výhodu v tom, že jasně vymezuje, ve kterém časovém období má smysl očekávat události, navíc je nastavení časového období velmi snadné, rychlé a intuitivní. Změna rozsahu je provedena posunutím posuvníku nebo změnou jeho délky pomocí interaktivních konců tohoto posuvníku. Nevýhodou tohoto řešení je složitější implementace z důvodu přepočítávání pozice a délky posuvníku na časové období. Vzhledem k benefitům vyplývajícím z tohoto řešení, volím změnu rozsahu pomocí interaktivního posuvníku.

Pro rychlé a snadné nalezení časového úseku, kdy se stalo velké množství událostí, bych ráda využila navigační časovou osu, obdobnou té, kterou využívá aplikace Tiki-Toki [9]. Tato navigační časová osa zobrazuje všechny události najednou jako malé body. Díky této navigační ose lze hned poznat v jakém časovém období se nenachází žádné události a naopak, kde se nachází hodně událostí. Aby bylo zřejmé, z které časové osy události jsou, budou body, které je představují, mít stejnou barvu jako časová osa, ke které patří.

Pro zkombinování interaktivního posuvníku a navigační časové osy využiji jejich překrytí, při kterém se vykreslí lišta s interaktivním posuvníkem a v ní body pro jednotlivé události. Návrh samotného posuvníku bez bodů pro události je na obrázku 5.5 pod číslem 7.

5.2 Stránka pro administrátory

V této části kapitoly popíši, jak bude vypadat webová stránka pro administrátory. Tato stránka bude umožňovat editaci událostí, tagů, aliasů, kategorií, ikonků a úpravu práv administrátorských účtů. Každá z těchto položek bude mít vlastní sekci na stránce, kde bude umožněno učinit potřebné operace. Administrátoři budou moci mezi těmito sekcemi přepínat dle potřeby.

Všechny tyto sekce budou umožňovat vyhledávání v již existujících záznamech, vybrat jeden z těchto záznamů a poté záznam upravovat. Dále bude možné vytvořit nový záznam s nutností vyplnit povinná pole. Záznamy také bude možné odstraňovat, pokud tím nebude porušena nějaká závislost např. tagy budou moci být odstraněny pouze pokud je nebude používat žádná událost. Rozložení sekce bude obsahovat vyhledávač záznamů a tabulku s výsledky vyhledávání v levé části obrazovky a v pravé části bude zobrazen formulář pro vyplnění a editaci údajů. Událostem a aliasům bude možné přiřadit tagy, což bude umožněno pomocí vyhledávače tagů a seznamu s již přiřazenými tagy, který bude umožňovat také tagy odebírat.

Sekce pro editaci ikonků bude mít rozdílné rozložení. Sekce bude rozdělovat ikonky na ikonky pro události a na ikonky pro tagy. Tomu se přizpůsobí i vzhled stránky, který bude zobrazovat v jedné části ikonky a vyhledávač pro tagy a v druhé části ikonky a vyhledávač pro události. V levé části stránky budou také zobrazeny aktuálně vybrané výchozí ikonky pro tagy i události. Každá ikonka bude mít pod sebou napsaný svůj název, tlačítka pro odstranění a přejmenování ikonky a tlačítko pro nastavení ikonky jako výchozí pro tagy nebo události.

Softwarový návrh

V této kapitole se popíše požadavky na systém a to na stránku Asterion Timelines i na stránku pro administrátory, poté popíše případy užití, doménový model a diagram tříd.

6.1 Funkční Požadavky

6.1.1 Zobrazení událostí na časové osy

Systém musí umět přehledně zobrazit události na časové osy, tak aby umístění události na časové ose odpovídalo datu, kdy se událost odehrála vzhledem k použitému měřítku. U každé zobrazené události půjde zobrazit její název, kdy se odehrála a její popis. U události, která je zobrazena na více časových osách, půjde na první pohled poznat, že se jedná o stejnou událost.

6.1.2 Nastavení vizualizace časových os

Uživatelé budou moci měnit barvu vybrané časové osy, počet časových os a jejich pořadí. Pořadí časových os půjde změnit buď ručně nebo pomocí algoritmu, který spočítá nejideálnější pořadí tak, aby časové osy byly co nejpřehlednější.

6.1.3 Výběr událostí k vizualizaci

Aplikace bude umožňovat výběr událostí, které se budou vizualizovat na časovou osu. Události budou vybrány pokud budou obsahovat uživatelem vybraný tag. Výběr tohoto tagu proběhne pomocí nepovinného výběru kategorie a pomocí vyhledání v názvu a popisku tagu na základě vstupu od uživatele.

6.1.4 Evidence událostí, tagů, aliasů a kategorií

Administrátoři budou moci spravovat události, tagy, aliasy a kategorie. To znamená, že budou moci vytvořit nový záznam a upravovat a odebírat již existující záznamy. U události bude evidován její název, popis, příslušné tagy a datum, kdy se událost odehrála, nebo datum jejího začátku a konce. U Tagu bude evidován jeho název, popis a kategorie.

6.1.5 Správa administrátorů

Systém bude umožňovat administrátorům, kteří na to mají dostatečná práva, přidávat nové administrátorské účty, rušit je a přidávat a odebírat jim právo na správu administrátorů. U administrátorů bude evidováno jejich přihlašovací jméno, email a haš jejich hesla.

6.1.6 Autorizace a autentizace administrátorů

Administrátoři budou moci spravovat data a ostatní administrátory pouze pokud budou přihlášení.

6.2 Nefunkční požadavky

6.2.1 Webová aplikace

Systém bude dostupný přes webový prohlížeč.

6.2.2 Grafické uživatelské rozhraní

Systém bude poskytovat grafické uživatelské rozhraní pro práci s vizualizací a pro editaci dat.

6.3 Případy užití

Případy užití jsem rozdělila na dvě části a to případy užití stránky Asterion Timelines, na které se bude nacházet vizualizace a jsou vyobrazeny na obrázku 6.2 a na stránku editoru, jehož případy užití jsou zobrazeny na obrázku 6.1.

6.3.1 Výběr tagu

Uživatel vybere kategorii tagu, který chce vybrat, pokud tuto kategorii zná. Pokud kategorii nezná nebo nechce vyhledávat podle kategorie, zvolí ve výběru kategorie položku všechny. Uživatel zadá celý název hledaného tagu nebo jeho část a systém uživateli zobrazí nabídku se všemi tagy, které obsahují v názvu nebo popisu uživatelem zadaný text a zároveň je v zadané kategorii. Uživatel poté vybere z nabídky ten tag, který si přeje vizualizovat na časové osy.

6.3.2 Změna zobrazeného časového období

Uživatel může změnit časové období, ve kterém se odehrály události, které se zobrazí na časové osy.

6.3.3 Prohlížení popisu události

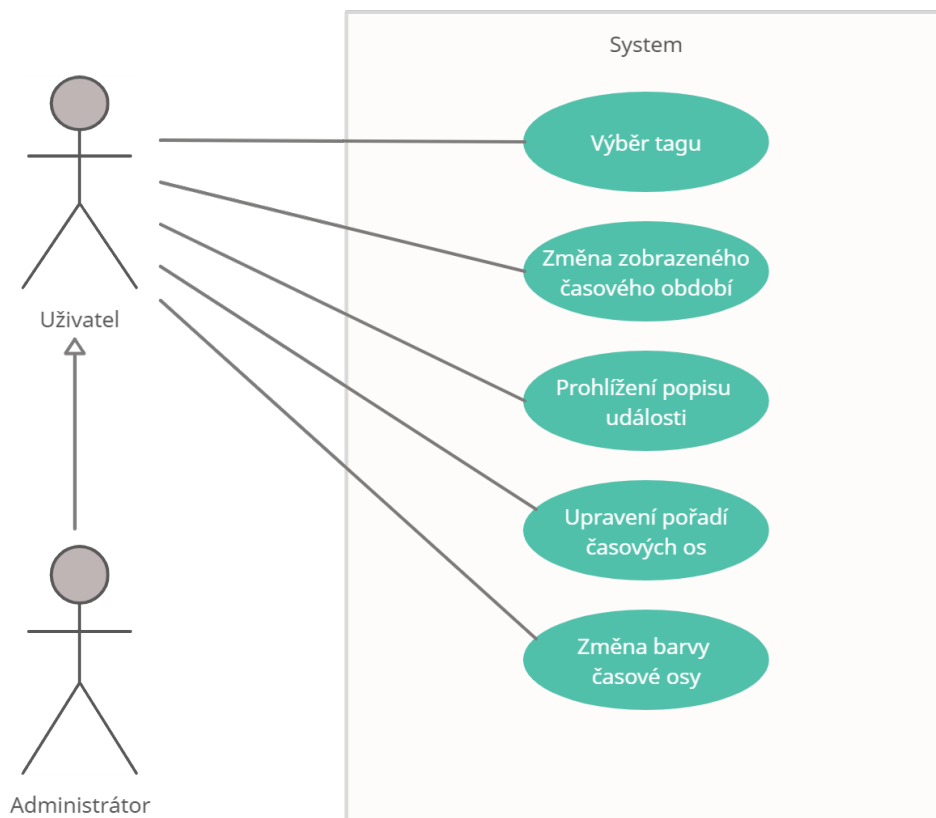
Uživatel si bude moci vybrat najetím myši událost, u které chce zobrazit její podrobnosti. Mezi podrobnosti patří název události, datum, kdy se událost udála, a popis, který bude podrobně popisovat průběh události.

6.3.4 Upravení pořadí a počtu časových os

Uživatel bude moci změnit pořadí časových os, které jsou zobrazené ve vizualizaci a jejich počet, pomocí přidání nové časové osy nebo odebráním některé z existujících.

6.3.5 Změna barvy časové osy

Barva časové osy bude nastavitelná na libovolnou barvu.



■ **Obrázek 6.1** Případy použití aplikace Asterion timelines

6.3.6 Přidání nových dat

Administrátoři budou moci přidávat záznamy nových událostí, tagů, kategorií a aliasů. Pro přidání záznamu musí administrátor vyplnit odpovídající formulář.

Ve formuláři pro přidání nové události bude povinné vyplnit název, popis a datum, kdy se událost odehrála. Mezi nepovinné údaje bude patřit ikonka a tagy. Pro vytvoření záznamu o tagu musí být vyplněn jeho název a kategorie, případně popis tagu. Kategorie obsahuje pouze povinný název a ikonku.

6.3.7 Úprava dat

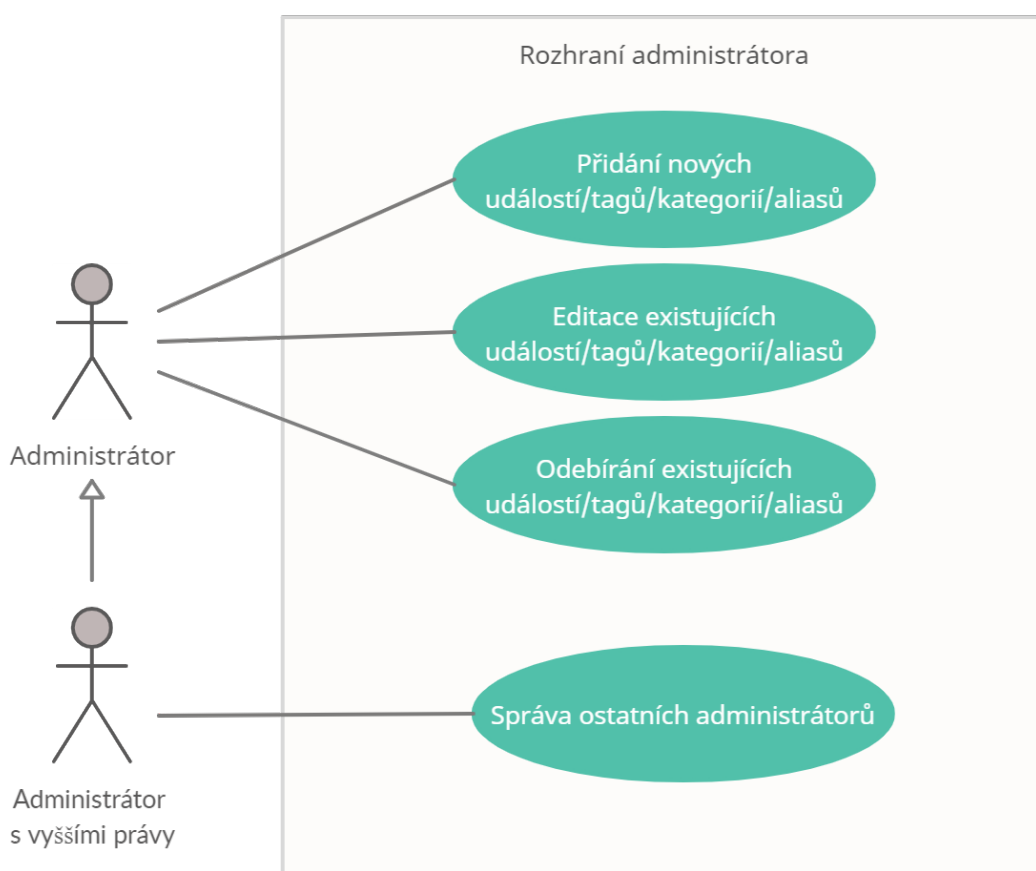
Administrátoři budou moci vyhledat již existující události, tagy, kategorie, aliasy a ikonky a upravit jejich data pomocí formuláře, tak aby zůstala zachovaná vyplněnost povinných částí formuláře.

6.3.8 Odebrání dat

Administrátoři budou moci odebrat existující záznamy událostí, tagů, kategorií i aliasů. To budou moci udělat tak, že vyhledají záznam a kliknou na tlačítko pro odstranění záznamu. Pro odstranění kategorie je nutné, aby žádný tag neměl tuto kategorii přiřazenou. Je-li odstraněn tag, nesmí být používán žádnou událostí.

6.3.9 Správa administrátorů

Administrátoři s dostatečnými právy budou moci vytvářet nové administrátorské účty, odebírat účty bez práv na správu administrátorů a přidávat práva již existujícím účtům. Pro vytvoření účtu je potřeba zadat email, uživatelské jméno a heslo.



■ **Obrázek 6.2** Případy použití Editoru

6.4 Doménový model

Doménový model na obrázku 6.3 zobrazuje základní entity systému, jejich atributy a vazby mezi sebou. Většina entit byla již dostatečně popsána v předchozích částech práce a proto je nebudu znovu popisovat.

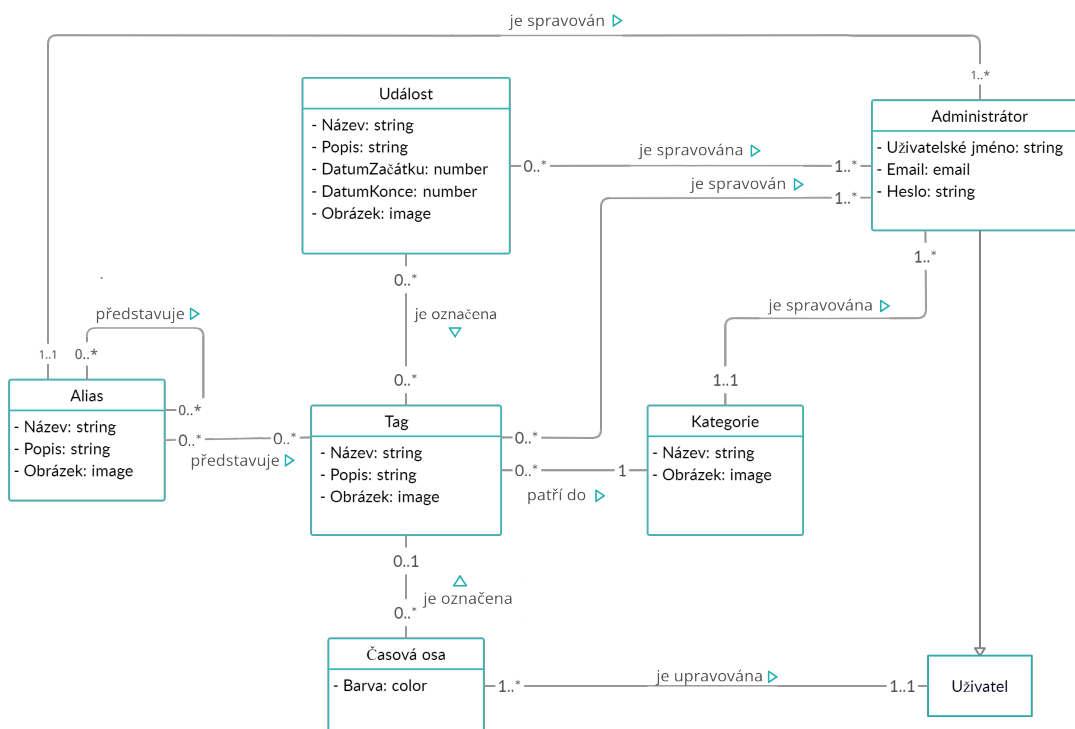
6.4.1 Alias

Alias je entita, která se v systému chová obdobně jako tag a stejně jako tag může představovat klíčové slovo události. Oproti tagu představuje nějaké klíčové slovo, které v sobě zahrnuje jiné tagy nebo aliasy.

Například pokud bychom v prostředí Asterionu chtěli zjistit, co se odehrálo ve Čtyřech královstvích, chtěli bychom, aby nám systém zobrazil všechny události označené tagy Almendor, Danérie, Keledor nebo Storabsko. Pokud by klíčové slovo Čtyři království bylo tag, tak bychom získali tyto události, musela by každá událost včetně těch, které se týkají pouze jednoho z království, být označena tagem Čtyři království. Pokud pro klíčové slovo Čtyři království použijeme alias, můžeme klíčové slovo dekomponovat a zobrazit výsledek hledání pro alias i jeho jednotlivé složky.

Alias také umožňuje hledat objekty, které jsou označovány více různými názvy, což bychom mohli využít např. pro zobrazení událostí z Mořského císařství, kterému se také říká Plavena.

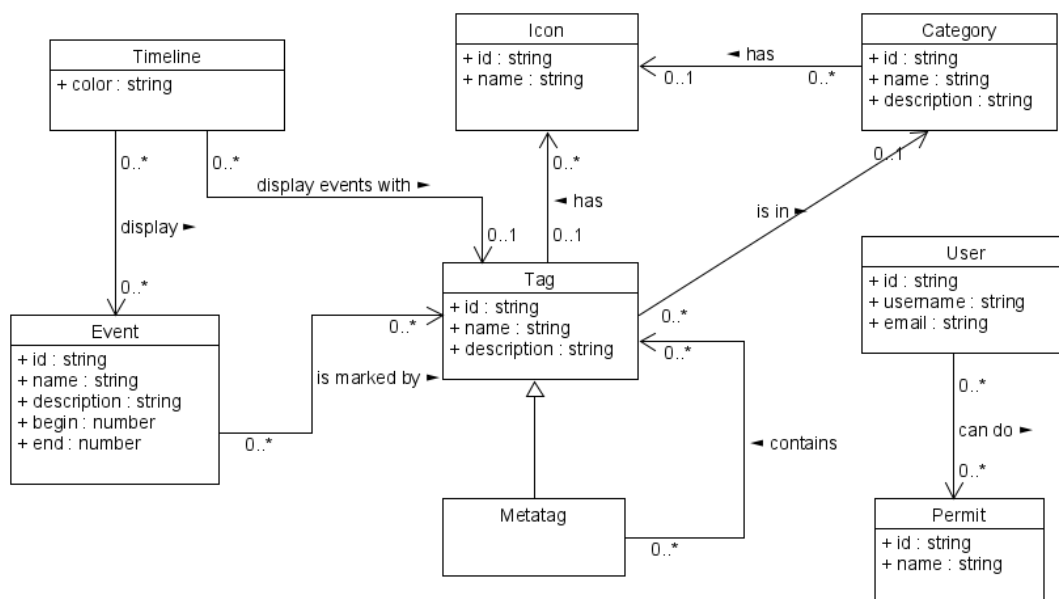
Všecké operace potřebné pro zpracování aliasu budou probíhat na straně serveru a webová stránka nebude schopná rozpoznat mezi tagem a aliasem, a proto se podrobněji aliasy nebudou zabývat.



■ Obrázek 6.3 Doménový model

6.5 Diagram tříd

V této části popíši všechny důležité třídy, které budou tvořit aplikaci. Nejdříve se budu zabývat třídami, které mají vizualizace a editor společné a poté popíši nejdůležitější části jednotlivých modulů.



■ Obrázek 6.4 Model tříd nesoucí data

6.5.1 Data

Veškerá data budou uchována v instancích tříd Timeline pro časové osy, Event pro události, Tag pro tagy, Category pro kategorie tagů, Metatag pro aliasy, User pro administrátorské účty a Icon pro ikonky, tak jak je to zobrazeno na diagramu těchto tříd 6.4.

6.5.2 Api

O veškerou komunikaci se serverem se budou starat třídy, které budou dědit ze třídy Service a rozdělí si mezi sebou správu dat podle většiny tříd popsaných v předchozí sekci. Tyto třídy jsou popsány na modelu 6.5.

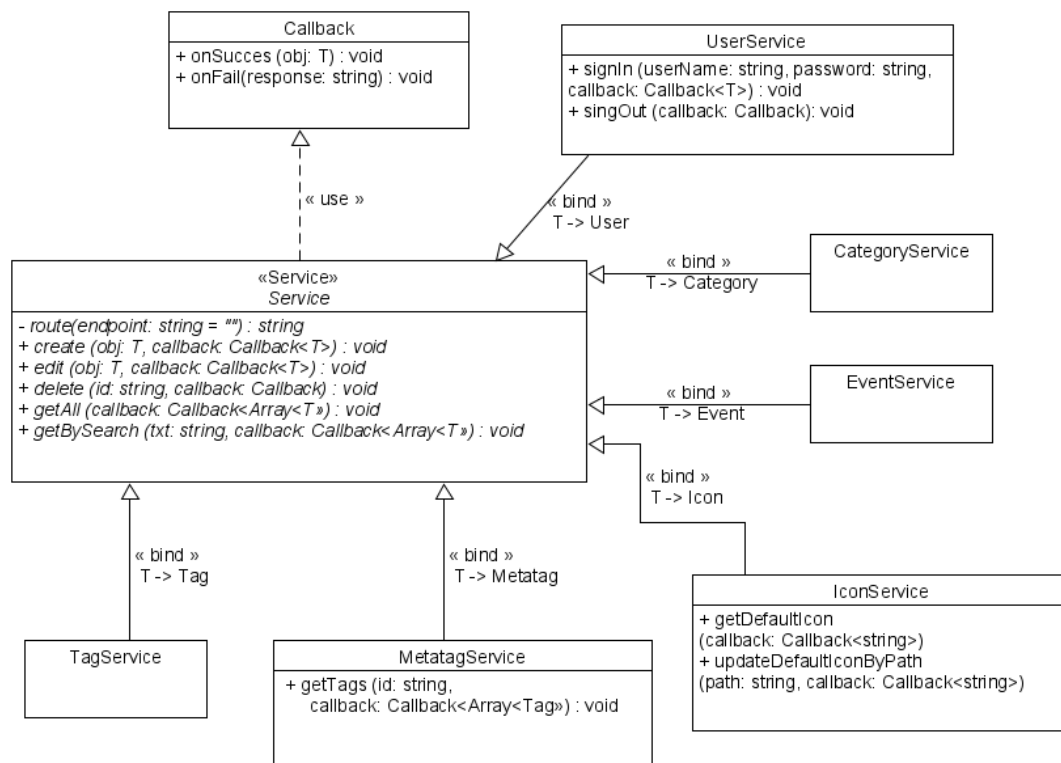
6.5.3 Vizualizace

Hlavní strukturu programu vizualizace budou tvořit čtyři třídy, první z nich se bude jmenovat App a bude zajišťovat komunikaci mezi ostatními třídami. Dalšími třídou bude TimelinesController, která bude poskytovat rozhraní pro práci s daty časových os (změna pořadí, přidání nové osy, odebrání existující osy apod.). Tu bude nejvíce využívat třída ControlPanel, která bude zajišťovat vstup od uživatele pro ovládací prvky. Vedle těchto tříd bude nejdůležitější třídou Renderer, která bude zajišťovat vykreslení vizualizace.

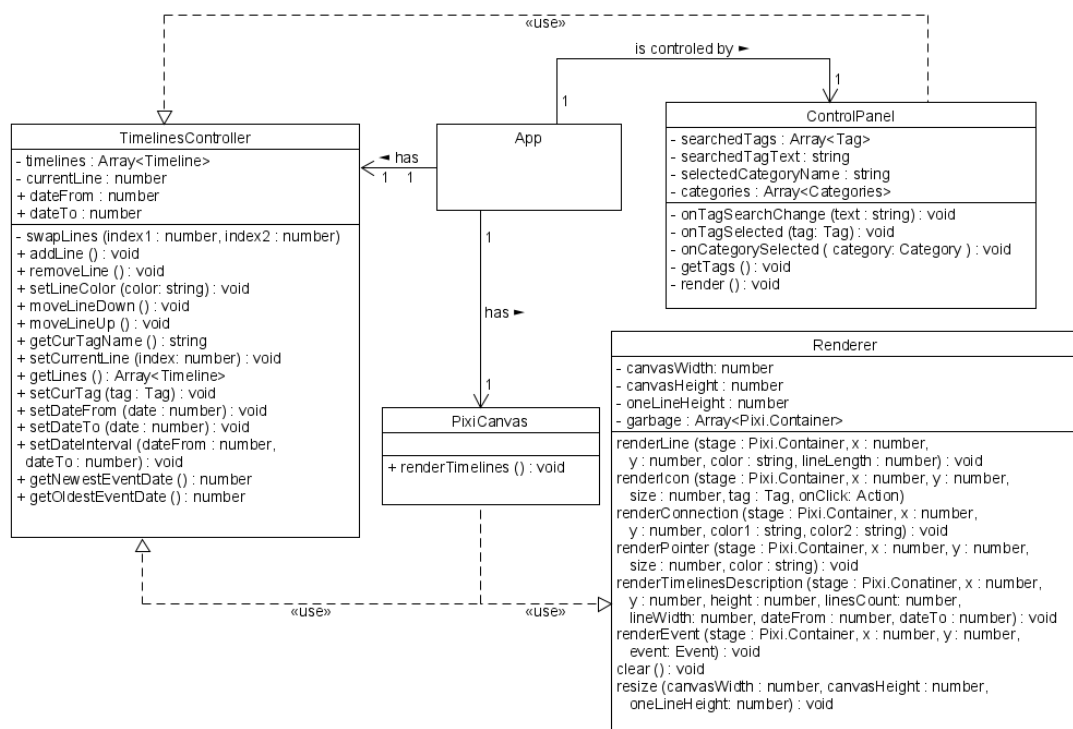
K tomu bude používat velké množství menších tříd, které budou řešit vykreslení vždy pouze jednoho konkrétního prvku vizualizace např. popisku události, ikonky tagu, popisku časové nebo ukazatel na aktuálně vybranou časovou osu.

Samotnou třídu Renderer bude používat třída PIXICanvas, která bude představovat plátno, na které se časové osy vykreslí.

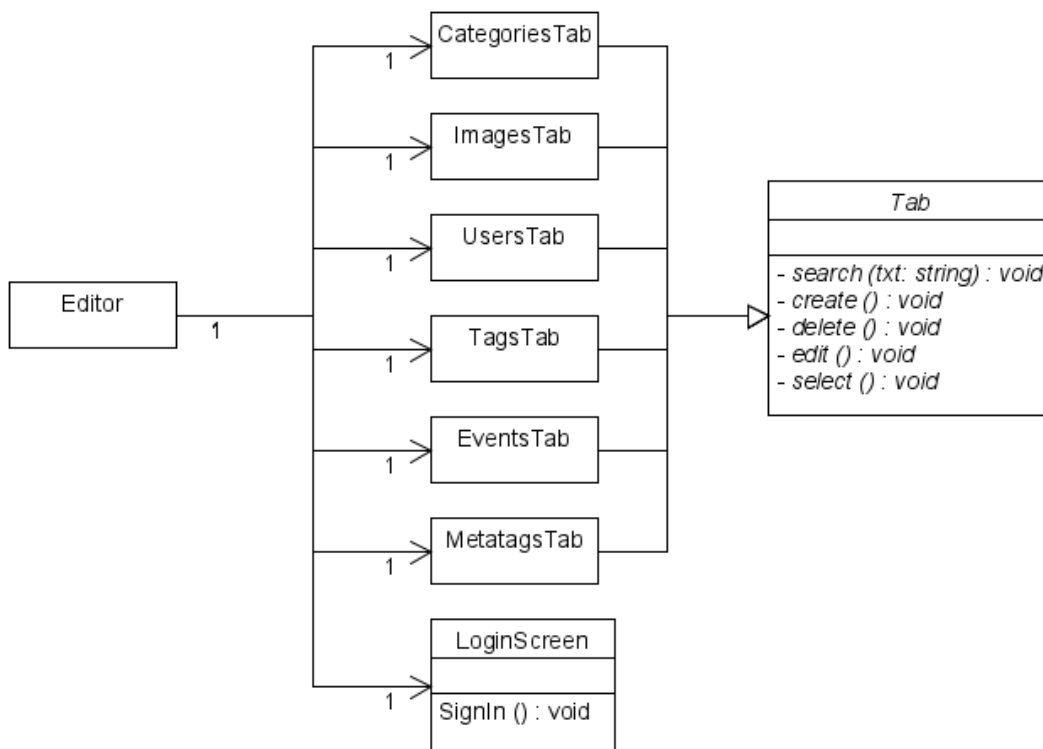
Všechny tyto třídy jsou podrobněji popsány na modelu 6.6.



■ **Obrázek 6.5** Model tříd pro komunikaci se serverem



■ **Obrázek 6.6** Model tříd tvořící základní strukturu programu



■ Obrázek 6.7 Základ modelu tříd editoru

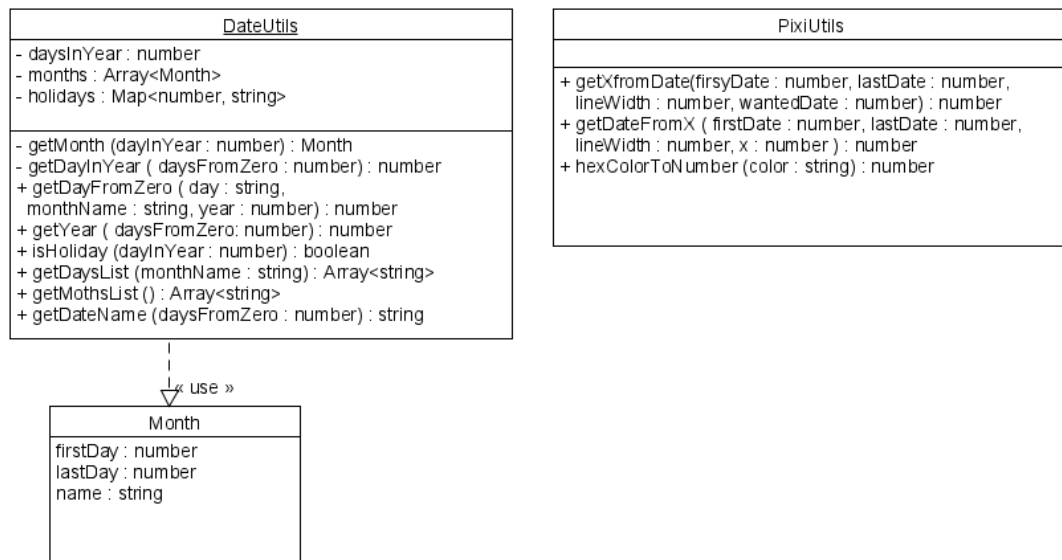
6.5.4 Editor

Hlavní strukturu editoru bude tvořit třída Editor, která bude umožňovat přepínat mezi jednotlivými sekcemi stránky a přihlašovací stránkou. Každou sekci bude tvořit třída, která dědí od třídy Tab, která představuje právě jednu sekci. Každá z těchto tříd bude vykreslovat rozhraní pro uživatele, které bude obsahovat vše potřebné pro vyhledávání záznamů, jejich editaci a vytváření nových záznamů přidělené části editoru. Všechny tyto třídy jsou zobrazeny na modelu 6.7.

6.5.5 Pomocné třídy

Pomocné třídy jsou statické třídy, které poskytují ostatním třídám nástroje pro opakující se výpočty, které jsou použity v různých částech kódu. Patří do nich třída DateUtils, která pomáhá s prací s časovými údaji a to např. s převodem číselně zadaného data do formátu používaném v Asterionu v Asterionském kalendářním systému a zpět.

Druhou třídou je PixiUtils, která nabízí rozhraní pro převod pozice na přímce (časové ose, posuvníku) na datum a zpět. Obě třídy jsou zobrazeny na diagramu 6.8.



■ **Obrázek 6.8** Model pomocných tříd

Implementace

Během implementace bylo implementováno vše, tak jak bylo popsáno v návrhu s výjimkou editace administrátorských účtů v editoru, který nebyl implementován. V této kapitole popíšeme krátce technologie, které byly při implementaci použity a poté všechny verze prototypu, které vznikly v rámci předmětů BI-SP1 a BI-SP2 a aktuální verzi.

7.1 Použité technologie

Základní použitou technologií je React [17]. Jedná se o knihovnu pro javascript, která umožňuje tvorbu uživatelského rozhraní.

Jednou z hlavních výhod knihovny je tvorba komponent, které spravují vlastní stav a na jeho základě vykreslují potřebný HTML kód. To umožňuje snadno reagovat na jakoukoli změnu v aplikaci. Komponenty lze použít vícekrát, takže nedochází k duplikaci kódu, čehož bylo využito například při implementaci odkazů, zobrazení ikoněk v sekci jejich editace v editoru nebo pro vstupy od uživatele ve všech formulářích v editoru.

Další použitou technologií je Typescript [18]. Typescript je jazyk postavený na Javascriptu, který přidává definice statických typů. To umožňuje popsat strukturu objektů a definovat typy proměnných. Díky tomu se lze snadno vyhnout chybám při typové kontrole.

Pro vizualizaci časových os byla použita knihovna PixiJS, která již byla popsána v kapitole řešerše technologií a byla v rámci této řešerše vybrána pro implementaci.

7.2 První verze

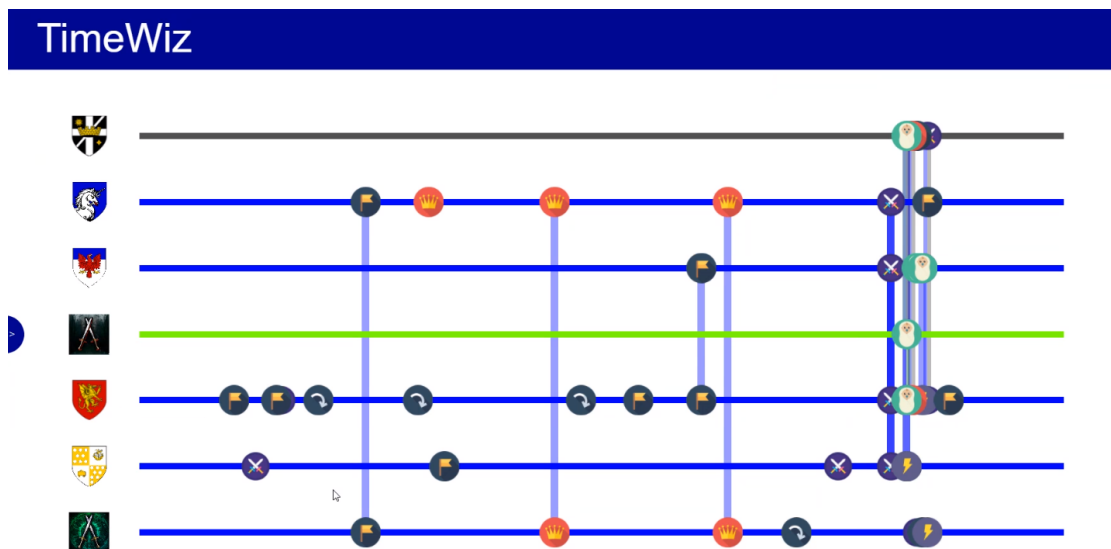
Aktuální implementace vizualizace se od první verze prototypu velmi liší. Původní verze umožňovala přepínat zobrazení mezi nastavením časových os a vizualizací os, což bylo při uživatelském testování velmi kritizováno. Pokud uživatel změnil nějaké nastavení časových os, tato změna se projevila až po přepnutí zobrazení na vizualizaci. Toto chování se jevilo uživatelům velmi matoucím.

První verze umožňovala vybrat více tagů pro jednu časovou osu, což se projevilo tím, že se na časovou osu zobrazili pouze události, které obsahovali všechny zadané tagy. Toto chování bylo podle testerů matoucí, protože nebylo jasné jakým způsobem jsou události vybrány.

Vizualizace první verze umožňovala změnit časové období pouze pomocí formuláře, ve kterém bylo nutné vybrat datum. Vizualizace také nerozlišovala mezi dny, ale pouze mezi roky, což vedlo k tomu, že se hodně událostí zobrazovalo na sebe.

Ikonky u časových os mohli měnit uživatelé sami z předem připraveného výběru. Tento výběr byl omezen pouze na dvacet ikoněk a proto nebylo možné vždy najít vhodnou ikonku pro tag.

Vzhled první verze byl velmi jednoduchý, nepoužíval moc příjemné barvy a celkově nebyl moc hezký. Na obrázku 7.1 lze vidět původní vizualizaci časových os.



■ Obrázek 7.1 První verze prototypu

7.3 Druhá verze

Druhá verze prototypu měla mezi testovanou skupinou oproti první verzi podstatně lepší odezvu. Již nepoužívala přepínání mezi zobrazeními a nastavení bylo umožněno pomocí lišty v horní části jako je tomu i v aktuální implementaci. Také umožňovala vybrat pouze jeden tag pro časovou osu, což je pro uživatele více intuitivní.

Oproti první verzi, tato verze přidávala k formuláři pro změnu časového období také ovládání pomocí šipek na klávesnici. Problémem tohoto posouvání byla špatná implementace, při které nedocházelo k uvolňování dat, která program vytvářel, což vedlo k tomu, že se aplikace po chvíli používání začala velmi zasekávat.

Pro usnadnění orientace byl do aplikace také přidán tutoriál. Druhá verze prototypu měla velmi pěkný vzhled, který jsem zachovala i do aktuální implementace.

Pro lepší orientaci na časových osách byl přidán popisek časových os, na kterém bylo zobrazena měřítko zobrazeného časového období.

7.4 Aktuální verze

Mezi hlavní rozdíly oproti předchozím verzím a aktuální patří posuvník, který je oproti formuláři i šípkám snazší a rychlejší na použití, ale oproti formuláři není tak přesný. S tím také souvisí nové zobrazení náhledu událostí v tomto posuvníku.

Druhým velkým rozdílem je odebrání možnosti výběru ikonky u časové osy uživatelem, místo toho lze ikonku nastavit v editoru napevno pro každý tag zvlášť, což může uživatelům ušetřit čas a také zobrazit větší množství různých ikon. Oproti předchozím verzím lze také nastavit barvu časové osy na libovolnou bez omezení předem připraveným výběrem.

Nová verze také přidává nápis se jménem události pod obrázek události na časové ose v případě, že nedojde ke kolizi s jiným nápisem.

Uživatelské testování

V této kapitole popíši jakým způsobem proběhlo uživatelské testování, kdo byl cílovou skupinou a výsledky tohoto testování.

Uživatelské testování proběhlo pomocí dotazníku, se zadáním několika jednoduchých úkonů s dotazy na obtížnost a intuitivnost potřebných úkonů.

8.1 Cílová skupina

Cílovou skupinou bylo několik hráčů RPG her, z nichž někteří hrají ve světě Asterionu. Odkaz na aplikaci i s formulářem byl k dispozici na serveru Asterionu na platformě Discord, kde kdokoliv chtěl, měl možnost aplikaci vyzkoušet, vyplnit dotazník a napsat mi anonymně zpětnou vazbu. Tato cílová skupina mohla otestovat pouze vizualizaci časových os. K tomuto testování proběhlo i testování osobně, při kterém několik testerů dostalo zadané stejné úkoly a vyplňovalo stejný formulář s možností osobního doplnění jejich pocitů z aplikace a navíc také otestovali i editor. Tito testeři byli hráči Dračího doupěte, ale bez znalosti Asterionu.

8.2 Návrh úkolů

Složitost úkolů se postupně stupňovala, aby bylo snazší postupně objevovat funkcionality aplikace. První úkoly byly zaměřené na zorientování se na časové ose pomocí zjišťování informací z připraveného zobrazení několika časových os. Pro zjištění potřebných informací se museli testeři orientovat na časových osách a být schopní, měnit zobrazené časové období a měnit vybrané tagy.

Druhá část testu byla zaměřená na úpravu časových os, jejich barvy, pořadí, odstranění osy a vytvoření nové časové osy. K těmto úkonům bylo nutné použít správné tlačítko na kontrolním panelu.

Ke každému z úkolů byla vždy možnost ohodnotit složitost tohoto úkonu pomocí výběru z následujících možností: snadné, přijatelné, těžké a nepodařilo se mi.

Závěrečná část testu se dotazovala na to, jak se testerovi pracovalo s aplikací a jak se mu líbí vzhled aplikace a zobrazení časových os.

8.3 Výsledek testování

Většina testerů se shodla, že nejtěžší při testování bylo splnit první instrukce kvůli nutnosti zorientovat se na časové ose. Po zjištění jakým způsobem časová osa funguje a jakým způsobem na ní vyhledávat jim již práce s časovými osami přišla snadná.

Vzhled aplikace byl ohodnocen ve většině případů velmi kladně. Většina testerů ocenila minimalistický a jednoduchý vzhled. Např. jeden z testerů napsal: „Použitá grafika je hezky jednoduchá. Nesnaží se to být přeplácaná, což pomáhá v přehlednosti.“ Během testování byla vizualizace vyzkoušena i na obrazovce s rozlišením 9600 x 4320 a zobrazení vypadalo i podle testera skvěle.

Při testování vizualizace byli zjištěny dva větší nedostatky. Prvním z nich je chybějící možnost přiblížit časovou osu dostatečně blízko, aby se události nepřekrývali. Tento nedostatek není s dosavadním množstvím událostí v databázi, tak znatelný, ale dle uživatelů s hlubší znalostí asterionské historie, lze očekávat velké množství událostí v krátkém časovém období blízko asterionské současnosti. Tento nedostatek byl napraven a aplikaci byla přidána možnost nastavit minimální a maximální datum, které lze na časových osách zobrazit. Díky tomu lze dosáhnout většího přiblížení ale také i oddálení. Pomocí posuvníku se lze pohybovat na časových osách pouze v zadaném rozmezí.

Druhým velkým nedostatkem je špatné zobrazení dlouhých popisků u událostí. Tento nedostatek nebyl vyřešen z důvodu využití dvou různých technologií pro vykreslování webové stránky. Popisek je vykreslován pomocí knihovny PixiJS, která může vykreslovat grafiku pouze na svém plátně. Pokud se tedy popisek nevejde na plátno, nelze ho překreslit přes HTML elementy a popisek vypadá jako uříznutý.

Během uživatelského testování byl editor otestován pěti testery v mé přítomnosti. Při tomto testování bylo objeveno několik drobných chyb. Mezi nimi byli např. povinnost vyplnění ikonky u tagu, přestože by ikonka být povinná neměla a opětovné načtení webové stránky při stisknutí klávesy backspace na klávesnici při psaní do číselného vstupu. Tyto drobné chyby byly opraveny.

Během testování testeři navrhli několik vlastních nápadů na nové funkcionality, kterými by bylo možné aplikaci rozšířit. Mezi tyto návrhy patří tlačítko Zpět, díky kterému by se dal vrátit stav aplikace do stavu před posledním vykonaným úkonem v aplikaci a přesun časových os pomocí chycení časové osy kurzorem a přetažením na cílovou pozici ve vizualizaci (drag and drop).

Také bylo navrženo umožnit uživatelům vytvářet vlastní data v editoru, která by byla dostupná pouze pro osobní použití nebo pro danou skupinu lidí (nejpravděpodobněji družinu hráčů). Na závěr mohli testeři anonymně napsat jakoukoli poznámku, kterou mi mohli říct, co se jim líbilo nebo naopak nelíbilo. Celý formulář můžete nalézt v přílohách.



Kapitola 9

Závěr

Hlavním cílem této práce bylo vytvořit webovou stránku pro vizualizaci časových událostí světa Asterion na časové osy. Sekundárním cílem bylo vytvořit editor, pomocí kterého bude možné jednoduše spravovat data pro tuto vizualizaci. Oba dva cíle se podařilo zrealizovat. V rámci práce bylo navrženo rozšíření editoru o možnost správy administrátorů, ale již nebylo implementováno.

Webové aplikace lze v rámci navazujících prací rozšířit přidáním možnosti tvorby dat uživateli, které by nemuseli být viditelné pro ostatní uživatele, ale pouze např. mezi uživateli ze stejné družiny (skupina hráčů, kteří hrají spolu). Vizualizace by šla rozšířit o přesun časových os pomocí uchopení a táhnutí myši (drag and drop) nebo případně posouvat tímto způsobem zobrazené časové období. Dalším vhodným rozšířením je optimalizace zobrazení pro malé obrazovky jako je např. mobilní telefon.

Výsledný systém je znovupoužitelný i v jiném kontextu než je Asterion. Se zaměněním kalendářního systému z asterionského na libovolný jiný lze webovou aplikaci použít i pro vizualizaci jakýchkoli jiných událostí např. s gregoriánským kalendářním systémem lze použít vizualizaci pro zobrazení historických dat České republiky.

..... Kapitola A

Formulář pro uživatelské testování

Asterion Timelines

Asterion Timelines (<https://asterion-timelines.cz>) je webová stránka určená pro vizualizaci historických událostí Asterionu na časové osy. Vyplněním tohoto formuláře a plněním zadaných instrukcí pomůžete s uživatelským testováním aplikace a případně opravit nalezené chyby a nejasnosti. Většina otázek v tomto formuláři obsahuje jednu instrukci a otázku na obtížnost jejího plnění.

1. Zjistěte kdy byla podepsána mírová smlouva mezi Almendorem a Hevreny. Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

2. Zjistěte kdy bylo rozděleno Velké království, tedy kdy vzniklo Storabsko. (Vyžaduje posun na časové ose). Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

3. Zjistěte kdy slaví narozeniny Bogaj, syn krále Krutji. (Vyžaduje změnu zvoleného tagu (štítku časové osy) jedné z časových os). Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

4. Přidejte novou časovou osu a zvolte na ní libovolný tag. Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

5. Změňte libovolně pořadí časových os. Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

6. Změňte barvu libovolné osy na jakoukoli jinou. Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

7. Odstraňte jednu z časových os. Jak těžké bylo instrukci splnit?

Označte jen jednu elipsu.

- Snadné
- Přijatelné
- Těžké
- Nepodařilo se mi
- Jiné: _____

8. Jak se vám aplikace ovládala?

Označte jen jednu elipsu.

- Bez problémů
- Některé funkce jsem dlouho hledal/a
- Bylo to chaotické
- Jiné: _____

9. Jak se vám líbilo zobrazení časových os?

Označte jen jednu elipsu.

- V pořádku
- Trochu nepřehledné
- Špatně se v tom cokoli hledá
- Nemám tušení, na co se koukám
- Jiné: _____

10. Jak se vám líbí vzhled webové stránky?

Označte jen jednu elipsu.

- Příjemný vzhled
- Pár věcí by potřebovalo doladit
- Vzhled se moc nepovedl
- Takovou hrůzu jsem ještě neviděl
- Jiné: _____

11. Pokud je něco co se vám na aplikaci líbilo a chtěli to zdůraznit, nebo naopak velmi nelíbilo, budeme rádi za veškerou zpětnou vazbu.

Místo pro vaše poznámky:

Obsah není vytvořen ani schválen Googlem.

Google Formuláře

Bibliografie

1. BREHMER, M.; LEE, B.; BACH, B.; RICHE, N. H.; MUNZNER, T. Timelines Revisited: A Design Space and Considerations for Expressive Storytelling. *IEEE Transactions on Visualization and Computer Graphics*. 2017, roč. 23, č. 9, s. 2151–2164. Dostupné z DOI: 10.1109/TVCG.2016.2614803.
2. INFO WE TRUST. *Creative Routines* [online]. 2021 [cit. 2021-03-16]. Dostupné z: <https://infowetrust.com/project/routines>.
3. ROSS INSTITUTE, MOEBIO LABS. *Ross spiral Curriculum* [online]. 2015 [cit. 2021-03-16]. Dostupné z: <https://spiral.ross.org/spiral/>.
4. ROCKCONTENT. *NBA Lockout* [online]. 2021 [cit. 2021-03-17]. Dostupné z: <https://visual.ly/community/Infographics/sports/nba-lockout>.
5. THE WORLD ANVIL TEAM. *World Anvil* [online]. 2017 [cit. 2021-03-19]. Dostupné z: <https://www.worldanvil.com>.
6. PROGENY SOFTWARE INC. *Timeline Maker Pro* [online]. 2021 [cit. 2021-03-22]. Dostupné z: <https://www.timelinemaker.com>.
7. TIMETOAST TIMELINES. *Timetoast* [online]. 2021 [cit. 2021-03-27]. Dostupné z: <https://www.timetoast.com>.
8. KNIGHT LAB. *Timeline JS* [online]. 2021 [cit. 2021-03-21]. Dostupné z: <http://timeline.knightlab.com>.
9. WEBALON LTD. *Tiki-Toki* [online]. 2021 [cit. 2021-03-27]. Dostupné z: <https://www.tiki-toki.com>.
10. BACKES, Marc. *Code Wall* [online]. 2021 [cit. 2021-04-20]. Dostupné z: <https://www.codewall.co.uk/html-css-and-javascript-explained-for-beginners/>.
11. MOZILLA; CONTRIBUTORS, individual. *MDN Web docs* [online]. 2021 [cit. 2021-04-20]. Dostupné z: <https://developer.mozilla.org/en-US/>.
12. YWORKS. *SVG, Canvas, WebGL? Visualization options for the web* [online]. 2018 [cit. 2021-04-20]. Dostupné z: <https://www.yworks.com/blog/svg-canvas-webgl>.
13. GOODBOY DIGITAL LTD. *PixiJS Documentation* [online]. 2021 [cit. 2021-04-20]. Dostupné z: <https://pixijs.download/release/docs/index.html>.
14. *Phaser documentation* [online]. 2021 [cit. 2021-04-20]. Dostupné z: <https://newdocs.phaser.io/docs/3.54.0>.
15. BOSTOCK, Mike. *D3.js* [online]. 2020 [cit. 2021-04-20]. Dostupné z: <https://d3js.org>.
16. TWO.JS. *Two.js documentation* [online] [cit. 2021-04-20]. Dostupné z: <https://two.js.org>.

17. FACEBOOK INC. *React* [online]. 2021 [cit. 2021-04-23]. Dostupné z: <https://reactjs.org>.
18. MICROSOFT. *Typescript* [online]. 2021 [cit. 2021-04-23]. Dostupné z: <https://www.typescriptlang.org>.

Obsah přiloženého média

	readme.txt	stručný popis obsahu média
	src	
	impl	zdrojové kódy implementace
	thesis	zdrojová forma práce ve formátu \LaTeX
	text	text práce
	thesis.pdf	text práce ve formátu PDF