

Maturitní práce

Materiály pro budoucí herní projekty

Autor: Jan Frídl

Oponent: Adam Přibyl

Vedoucí Práce: Marie Souchová

Jan Frídl 4.A

Smíchovská Střední Průmyslová
Škola a Gymnázium 2021-2022

MATURITNÍ PRÁCE

Školní rok: 2021/2022

Vedoucí práce: Marie Souchová

Oponent: Adam Přibyl

Zadání: Materiály pro budoucí herní projekty

Účel projektu: Vytvoření užitečných materiálů pro usnadnění budoucího vývoje

Výstup (výstupy) projektu: Software

Obsah: (rozepsané úkoly)

1. Proveďte analýzu problému a navrhněte postupy pro splnění zadání
2. Sestavte harmonogram prací a hodnotící list s bodovanými úkoly
3. Proveďte vlastní řešení projektu dle schváleného harmonogramu
4. Proveďte závěrečné zhodnocení projektu
5. Odevzdejte závěrečnou zprávu ve formě dokumentu v programu Word
6. Odevzdejte prezentaci projektu v programu PowerPoint
7. Proveďte veřejnou prezentaci svého projektu

Maturitní projekt bude mít teoretickou a praktickou část. V teoretické části odevzdáte vytisknou závěrečnou zprávu a případné další výstupy, plynoucí z Vašeho projektu. Závěrečná zpráva, podklady pro prezentaci a případné další výstupy budou rovněž na připojeném nosiči CD. Součástí projektu je závěrečná veřejná prezentace projektu před třídou a dalšími návštěvníky prezentace.

V praktické části budete svůj projekt obhajovat před maturitní komisí.

Datum a podpis vedoucího práce:

Hodnotící list

název maturitní práce	příjmení a jméno			třída 4.A	školní rok: 2021 - 2022
Materiály pro budoucí herní projekty	Jan Frídl			oponent	Adam Přibyl
	termín	datum	max. body	hodnocení	poznámka
Úvod do PRO, pravidla PRO	1/35	5.9.			
Zadání projektu	2/36	12.9.			
Kontrola zadání projektu	3/37	19.9.			
Příprava analýzy	4/38	26.9.			
Odevzdání analýzy	5/39	3.10.	5 bodů	5 bodů	
Příprava harmonogramu	6/40	10.10.			
Odevzdání harmonogramu	7/41	17.10.	5 bodů	5 bodů	
Práce na pohybu hráče a zákl. prostředí	7/42	24.10.			
Vývoj pomocných nástrojů	9/43	31.10.			
Odevzdání 1. výstupu	10/44	7.11.	15 bodů	15 bodů	
Programování uchopení předmětů	11/45	14.11.			
Programování logiky pásů	12/46	21.11.			
Práce na pohonu a odpružení pásů	13/47	28.11.			
Odevzdání 2. výstupu	14/48	5.12.	20 bodů	20 bodů	posunutí termínu na 13.12.
Programování děl	15/49	12.12.			
Práce na hlavním menu	16/50	19.12.			
Vánoční prázdniny	17/51	26.12.			
Odevzdání 3. výstupu	18/52	2.1.	15 bodů	15 bodů	posunutí termínu na 9.1.
Nahrazování dočasných modelů	19/1	9.1.			
Vylepšování vzhledu	20/2	16.1.			
Odevzdání 4. výstupu	21/3	23.1.	10 bodů	10 bodů	
Upravování kódu	22/4	30.1.			
Ladění chyb	23/5	6.2.			
Odevzdání 5. výstupu	24/6	13.2.	10 bodů	10 bodů	
Práce na plakátu	25/7	20.2.			
Odevzdání plakátu	26/8	27.2.	5 bodů	5 bodů	
Jarní prázdniny	27/9	6.3.			
Odevzdání projektu a závěrečné zprávy včetně anotace v AJ	28/10	13.3.	5 bodů		
Odevzdání prezentace k MP	29/11	20.3.	10 bodů		
Prezentace maturitního projektu	30/12	27.3.			
Prezentace maturitního projektu	31/13	3.4.			
Prezentace maturitního projektu	32/14	10.4.			
Prezentace maturitního projektu	33/15	17.4.			
Prezentace maturitního projektu	34/16	24.4.			
Celkem bodů za průběžnou práci			100 bodů		
Celkové hodnocení:					
94 - 100 bodů ----- výborný					
85 - 93 bodů ----- chvalitebný					
71 - 84 bodů ----- dobrý					
51 - 70 bodů ----- dostatečný					
0 - 50 bodů ----- nedostatečný					
Pokud žák neřeší libovolný bod, je práce hodnocena jako nedostatečná.					
Hodnocení 2/IV					Hodnocení 3/V
Hodnocení 4/IV					podpis vedoucího práce

Čestné prohlášení

Prohlašuji, že jsem maturitní práci vypracoval samostatně pod vedením Mgr. Marie Souchové. V práci jsem použil informační zdroje uvedené v seznamu použité literatury a internetových stránek.

Praha

Podpis celým jménem

Anotace

Cílem mé maturitní práce bylo vytvoření podkladů a materiálů souvisejících s vývojem her. Tyto podklady by potom mohly být mnou použity k usnadnění budoucího vývoje. Vybral jsem si toto téma, jelikož jsem se minulý rok zabýval tvorbou hry v rámci studentského projektu a jelikož je to něco, čím bych se chtěl dále zabývat.

Většina podkladů je buďto ve formě C# skriptů, určených pro herní engine Unity3D, anebo ve formě 3D modelů vyrobených v Blenderu.

Výstupy mojí práce se skládají ze spustitelného build souboru a z dokumentace, která popisuje vývojařské nástroje a postup v práci.

Annotation

The goal of my maturita project was to create source material related to game development. This source material could then be used by me to ease my future development. I have chosen this topic since I have worked on a game last year as a part of my student project and it is a field that I would like to keep exploring.

Most of the source material is in the form of C# scripts made to work in the Unity3D game engine or in the form of 3D models created using Blender.

The turned in work consists of an executable build file and documentation describing the developer tools as well as the progress made.

Obsah

1	Úvod	2
2	Analýza maturitní práce.....	3
2.1	Popis úkolu.....	3
2.2	Popis stávajícího stavu	3
2.3	Výběr vhodných prostředků pro řešení úkolu	3
2.4	Výběr řešení a výstupů.....	4
2.5	Stanovení dílčích úkolů.....	5
3	Popis vlastního řešení práce	6
4	Závěr	17
5	Seznam zdrojů	17
6	Seznam obrázků.....	17
7	Seznam příloh	18

1 Úvod

Když jsem si vybíral téma pro svoji maturitní práci, byl jsem si prakticky jistý oblastí práce, tedy herním vývojem, ale nemohl jsem přijít na to, co přesně bych chtěl udělat. Po nějaké době přemýšlení mne napadlo, že bych si mohl ulehčit svoji budoucnost a vytvořit si nějaké základy pro mé budoucí projekty. Při přesnějším formulování toho, čeho bych chtěl v rámci maturitní práce, se spojila moje dlouholetá záliba v tanky a obrněná vozidla a fascinace fyzickými simulacemi a rozhodl jsem se že vytvořím modulární systém pro tvorbu pásového podvozku pro tank spolu se základním ovládacím systémem. Tento systém by umožnil relativně jednoduché vygenerování podvozku pro téměř jakýkoliv typ tanku či obrněného vozidla a také by umožnil jeho následnou kompletní simulaci v reálném čase, přesněji tedy simulaci jednotlivých článků pásu a jejich poškození, odpružení podvozku a pohon tanku.

Během minulého roku jsem si dokázal osvojit základy herního vývoje a jaké různé nečekané problémy se mohou během práce objevit. S vytvářením simulací jsem ale neměl příliš velké zkušenosti, a tak na mě čekalo prozkoumání obrovského pole a nalezení nespočetně frustrujících a často zdánlivě nepochopitelných problémů. Věřil jsem však, že díky mým dlouhodobým technickým zkušenostem budu schopen svého cíle dosáhnout a otázkou zůstávalo pouze kolik času mi tato práce zabere. Jak se ukázalo, bylo ho opravdu hodně.

2 Analýza maturitní práce

2.1 Popis úkolu

Mým cílem pro maturitní práci je vytvořit interaktivní 3D prostředí s nádechem muzea ve kterém si hráč bude moci prohlédnout a vyzkoušet mechaniky vyvinuté pro mojí budoucí hru zaměřující se na stavbu a vývoj tanků. Prostředím se hráč bude moci volně pohybovat a u většiny mechanik se mu dostane mého komentáře ohledně důvodu existence určité mechaniky, vývoje a také různých nedostatků. Finální produkt bude tedy cílen spíše na další vývojáře, či na hráče, které zajímá, jak hry fungují za jejich zády.

2.2 Popis stávajícího stavu

Hlavním zaměřením této práce bude realistická simulace tankových pásu v reálném čase a jejich dynamické vytvoření. Přestože se na internetu dá nalézt mnoho příkladů právě těchto simulací, neexistuje příliš mnoho materiálu, který se zabývá jejich tvorbou, většinou se dají nalézt pouze hotové produkty k zakoupení, či videa ukazující výslednou simulaci. K dosažení tohoto cíle budu tudíž muset využít hlavně svých zkušeností s vývojem her kterých jsem nabil minulý rok při tvorbě studentského projektu a také svých programovacích zkušeností, kterých mám díky různému experimentování větší počet. Zároveň také budu využívat různých vývojářských fór či tutoriálů abych pochopil či vymyslel jednotlivé součásti simulace a vyrobil nástroj který mi umožní jednoduše vytvořit funkční pásy dle zadaných požadavků.

2.3 Výběr vhodných prostředků pro řešení úkolu

Pro splnění mého cíle budu potřebovat několik druhů programů. Budu potřebovat software který mi umožní sledovat můj postup. Dále budu potřebovat program který mi umožní vytvářet modely pro mé mechaniky a samozřejmě prostředí ve kterém tyto mechaniky budu moci vytvářet a programovat. Také musím vyřešit problém přenositelnosti souborů, jelikož hodlám pracovat na více zařízení.

Management software

Tabulka 1 - Management software

Program	Dostupnost	Funkce	Znalost	Součet	Pořadí
Trello	10	7	8	25	1
Asana	10	8	0	18	2
Hubstaff	6	10	0	16	3

Bodování 1-10 (10 nejlepší)

Zde volba nebyla příliš složitá, a to i přesto, že na trhu existuje nespočetně aplikací tohoto druhu. S Trellem mám již delší zkušenosti, a tak nad ostatními funkcionálně lepšími alternativami lehce zvítězilo.

3D Engine

Tabulka 2 - 3D engine

Engine	Dostupnost	Funkce	Znalost	Komunita	Součet	Pořadí
Unity 3D	10	8	9	9	36	1
Unreal Engine	10	9	0	6	25	2

Bodování 1-10 (10 nejlepší)

Pro svůj herní engine jsem opět vybral Unity, jelikož jsem v s ním pracoval předešlý rok a také jelikož má o mnoho více výukového materiálu a podpory (vyjádřeno jako „Komunita“)

2.4 Výběr řešení a výstupů

Jelikož je můj původní výkonný počítač neopravitelně poškozen a díky čipové krizi nemám k dispozici dostatečné finance na nákupu náhrady, nemám příliš na výběr při řešení práce. Finální produkt bude muset být úsporný a efektivní, jelikož bych na něm jinak nemohl pracovat. Toto bude hlavní limitující faktor. Výstupy poté budou ve formě „buildnuté“ aplikace spolu s dokumentem, ve kterém budou předvedeny vývojářské nástroje nepřístupné hráči v aplikaci a také přehled přidaných věcí.

2.5 Stanovení dílčích úkolů

1. Vymyšlení všech realizovatelných mechanik
2. Naprogramování mechanik
3. Návrh a tvorba finálního prostředí
4. Návrh a tvorba grafiky mechanik
5. Tvorba doprovodných komentářů
6. Upravování a vylepšování práce
7. Dokončení práce

3 Popis vlastního řešení práce

Když jsem poprvé začal dělat na své maturitní práci, došlo mi, že jsem již velmi dlouho nevytvářel žádné 3D modely, a že jsem nikdy předtím nevytvářel modely určené pro hru. Toto byl pro mě problém, jelikož modely určené pro herní prostředí bývají nározdí od ostatních lépe optimalizované a při jejich tvorbě se dbá na správnou strukturu topologie. Jelikož jsem v té době neměl žádný výkonný hardware, musely být mé modely velmi optimalizované, a tak jsem se rozhodl naučit se a využít několik technik které by měly snížit grafickou zátěž mého projektu, což by mi umožnilo využít více výkonu pro část simulační. Základní technikou byl tzv. low-poly styl modelů. V tomto stylu jsou modely záměrně tvořeny vzhledově hrubé a často vypadají jako by pocházely z 20 let starých her. Aby toho nebylo málo rozhodl jsem se také využít absolutní minimum textur a používání jedné textury pro více objektů, díky čemuž se do paměti vždy načítalo několik velkých textur, které v ní mohly zůstat po delší dobu namísto neustálého načítání a uvolňování mnoha textur pro různé objekty.



Obrázek 1 - Cvičný "low-poly" model

Po obnovení a nabítí nových modelovacích zkušeností bylo na čase začít pracovat na technické stránce. Ze všeho nejdřív jsem si chtěl vytvořit hráčskou postavu a pohybový systém, pomocí kterého by tato postava šla ovládat. Tato část nebyla sama o sobě příliš důležitá pro vytvoření simulovaných tanků, ale byla důležitá z hlediska vývojového, jelikož jsem pro zprovoznění svého pohybového systému musel vytvořit několik nástrojů které jsem využíval po zbytek své maturitní práce. Také mi práce na tomto pohybovém systému umožnila se znova zorientovat v herním programování, jelikož byla více podobná zadáním, které jsem zpracovával v minulosti.

Jeden z velmi důležitých nástrojů, který jsem během programování pohybového systému vytvořil byl Cylinder Collider, neboli generátor válcových kolizí. Tento nástroj

jsem musel vytvořit, jelikož herní engine, který jsem používal ke své práci v základu neobsahoval podobný nástroj kvůli složité rovnováze mezi přesností a výpočetní složitostí výsledného produktu. Čirou náhodou jsem základy pro podobný systém vyvinul minulý rok při práci na studentském projektu, a tak jsem se rozhodl tyto základy znova prostudovat a využít je ve svém novém nástroji.

Původně jsem se pokoušel vymyslet si vlastní vzorce které by mi umožnily rychlejší generování kolizního válce. Jak se ale ukázalo, mé vzorce byly nepřesné při generování zjednodušených válců, které měly velmi omezený počet stran pláště, a stávali se přesnými až při vytváření objektu který se blížil svojí přesnosti k válci. Zde ale nebylo limitujícím faktorem řešení matematických vzorců, ale samotné generování mnoha částí, ze kterých se výsledný válec skládal, a tak jsem od svých vzorců upustil a využil jsem opravdových, ozkoušených matematických rovnic. Po tomto rozhodnutí se již žádný velký problém neobjevil a nástroj se mi povedlo dokončit. Tento nástroj mi umožnil stanovit si počet stěn válce a velikost na jeho různých osách a následně dle mých požadavků válec vždy bez problémů vygeneroval. S tímto výsledkem jsem byl velice spokojen, a tak nastal čas na to posunout se dál v mojí práci.

Jelikož jsem původně čekal že mi vytvoření takového systému zabere většinu výstupu, byl jsem velice překvapen, když jsem ho dodělal během týdne, a tak mi zbylo hodně času na vytvoření čeho chci.

Na mém novém nástroji, a na kolizích celkově, mě iritovala jedna věc – během hraní nešly rychle a jednoduše zobrazit. Abych je viděl, musel jsem se vždy přepnout do editovacího módu, což ale znamenalo že nemohu ovládat svoji postavu. Rozhodl jsem se tedy, že vytvořím systém, který mi umožní tyto kolize vidět v jakékoli situaci. Byl to poněkud ambiciózní plán, ale jelikož jsem v tu dobu měl hodně času tak jsem neviděl důvod proč se o něj nepokusit.

Začal jsem tím, že jsem si pro každý neviditelný kolizní objekt vygeneroval jeho viditelný protějšek, který sám kolize neměl. Následně jsem musel pouze nastavit identické pozice a všechny nově vytvořené protějšky přidat do seznamu. Toto mi později umožnilo přepínat mezi nově vygenerovanými protějšky a normálními viditelnými objekty.

I když mi fungovalo přepínání objektů zbývalo překonat další problém který se nečekaně objevil. Můj generátor válců vytvářel objekty za běhu hry, ale seznam pro

objekty s kolizemi se vytvářel pouze při spuštění. To způsobilo nesrovnalosti a nefunkčnost mého nového systému. Musel jsem tedy vytvořit další systém dedikovaný k přijímání a předávání nových událostí. Tento systém přijímal události od mého generátoru válců a předával je dál k mému vizualizačnímu systému, který díky nim mohl aktualizovat seznam, když byl nějaký objekt vytvořen či smazán.

Po úspěšném vytvoření nástrojů jsem se rozhodl opět odbočit zpět k hráčské postavě. Věděl jsem, že v mojí hře budou fyzické objekty, a chtěl jsem, aby s nimi mohl hráče nějak interagovat, tudíž mě napadlo implementovat systém umožňující uchopení a přenášení předmětů.

Začal jsem tím, že jsem označil vybrané předměty jako uchopitelné, aby je můj skript mohl rozpoznat od normálních fyzických objektů, jako jsou vozidla. Následně jsem si vytvořil určité limity pro maximální váhu, kterou hráč může uchopit a maximální vzdálenost ve které hráč může být, aby mohl předmět zvednout. Když jsem toto vše měl hotové stačilo mi pouze přidat akční klávesu při jejímž stisku bude detekován nejbližší objekt ve směru hráčova pohledu. Pokud je objekt v uchopitelné vzdálenosti a má dostatečně nízkou váhu, bude nyní hráčem držen. Hráč se s tímto objektem poté může volně pohybovat a opakováním stisknutím akční klávesy jej může upustit. Pokud se objekt někde zaseknut a hráč se od něj pokusí odejít, bude objekt také upuštěn.

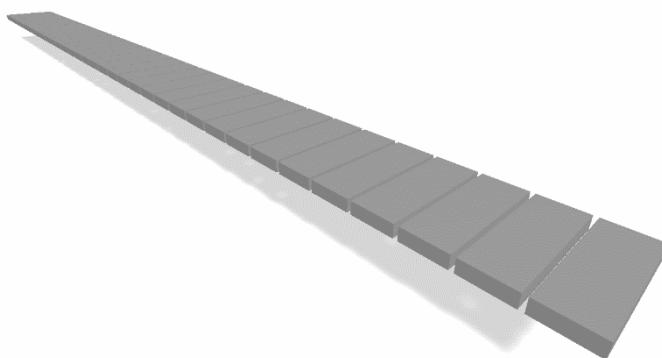
Tento systém má ale menší chybu. Umožňuje totiž hráči zneužít chybějících fyzických opatření, které jsem neimplementoval kvůli nedostatku času a z důvodu práce na důležitějších elementech. Přesněji tedy umožňuje vzít objekt a natlačit ho pod sebe. Jelikož se objekt pokouší udržet v definované vzdálenosti od hráče, je tlačen silou směrem k hráči, zároveň se ale nemůže přiblížit více protože má kolizi s hráčem, a tak je tato síla přenášena na hráče a překonává gravitaci která na něj působí, což způsobí že hráč vzletí do vzduchu, kde zůstane, dokud předmět neupustí, anebo nezmění jeho pozici tak, aby nebyl pod ním.

Čas utíkal, a tak jsem se začal soustředit na hlavní část projektu, tedy automatizované generování pásů. Měl jsem přibližně vymyšleno, jak by celý systém měl fungovat, ale nepočítal jsem kolik problémů se při vývoji objeví, a kolik času mi celá část zabere.

Než abych se po hlavě pouštěl do udělání celého systému naráz, rozhodl jsem se, že by nebyl špatný nápad rozdělit celý generační systém na několik pod-systému které bych po jejich dokončení mohl integrovat do jednoho kompletního systému. V závěru jsem

byl velice rád že jsem toto udělal, protože i přes odložení termínu mi nezbylo příliš mnoho času než na kompletní základy.

První z částí tohoto systému bylo podrobnější prozkoumání procedurálního generování a vytvoření skriptu který by vygeneroval prostý pás položený v rovné linii. Systém jsem dělal primitivní schválně, protože jsem chtěl vyzkoušet procedurální propojování fyzických objektů, jelikož jsem do tohoto bodu nic podobného nedělal. Jak se ukázalo, pod povrchem tohoto zdánlivě jednoduchého úkolu se skrývalo mnoho problému které jsem vyhlazoval až do konce maturitní práce.



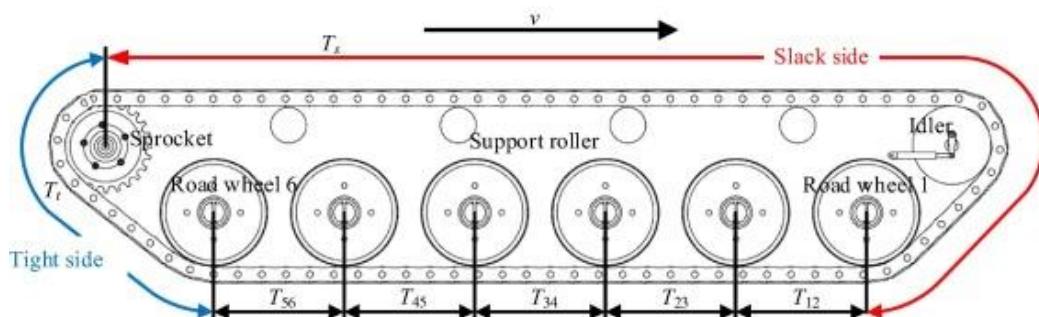
Obrázek 2 - Primitivní vygenerovaný pás

Samotné vygenerování a spojení jednotlivých článků nebylo problémové, a za krátkou dobu jsem měl hotový skript schopný vygenerování pásu s libovolným modelem a lehce určitelnou délkou pásu a rozestupy mezi články. Když jsem ale spustil svůj projekt v Unity, začaly se projevovat mnohé problémy. Prvním z nich byla náhodná a zprvu nevysvětlitelná disintegrace pásu s více než 20 články. Jak se totiž ukázalo, Unity z neznámého důvodu nastavovalo těžitště nově vytvořených článků kompletně mimo jejich fyzický rozsah a inerční hodnoty byly kompletně nesmyslné. Když se pak tyto 2 fakty při spuštění simulace zkombinovaly, došlo velmi rychle k extrémní oscilaci a následné disintegraci celého pásu. Myslím si, že je zřejmé, že takto se reálné pásy nechovají, a tak jsem začal hledat způsob, jak se tohoto chování zbavit. V tu dobu jsem ještě netušil, co přesně toto roztrhání způsobuje, a trvalo mi dobrých pár hodin hledání na internetu, než jsem vůbec našel zmínku která byla relevantní mému problému na zapadlé fóru. Naštěstí na fóru bylo popsáno přesně to, s čím jsem se potýkal i spolu s řešením a tak jsem se mohl vrhnout na řešení problému dalšího – roztahování a celková nízká pevnost pásu. Toto byl problém více komplikovaný, jelikož z větší části byl způsoben nastavením fyziky v samotném vývojovém prostředí a neexistovalo na něj

žádné plné řešení. Vylepšení chování pásu znamenalo nespočet hodin ztrávených opatrným upravováním hodnot, testováním a pozorováním výsledků, a to stále dokola. Aby toho nebylo málo, spouštění simulace trvalo často minutu díky mému slabému hardwaru a často se stalo, že projekt jednoduše zamrzal a musel být restartován. I když po dlouhém balancování nebyl výsledek ideální, nebylo moc času na to ho dále vylepšovat a prozatím musel postačit.

Nyní byl čas na systém druhý, tedy generování pásu po obvodu kol tanku. Pro funkčnost této části musela být fyzická kola převedena na kružnice a mezi sousedními koly došlo k vytvoření tečen po vnějším obvodu soustavy kol. Nelze ale spojit jakékoli sousední kolo, jelikož reálné pásy vedou po venkovním obvodu a postupují z jednoho kola na druhé. Pokud bych tedy spojoval nejbližší kola, mohlo by dojít k různým chybám, jako propojení spodních a horních kol, či nesmyslné změny směru pásu. Musel jsem tedy identifikovat a definovat základní typy kol a interakce mezi nimi.

Obvyklý tank má minimálně 3 typy kol. Nejdůležitější z nich je hnací kolo (Sprocket), které je jako jediné poháněno motorem a udává celý pás do pohybu. Následně typem je poté kolo napínací (Idler). Toto kolo se stará o správné napětí pásu, aby nebyl moc volný či moc pevný. Posledně jsou tu kola pojazdová (Road wheel), na kterých spočívá celá váha tanku a která se starají o odpružení nárazů. Tanky také většinou mají kladková kola (Return / Support roller) z důvodu snížení napětí v pásu. Přestože tato kola nejsou nezbytná pro fungování pásového vozidla, rozhodl jsem se je do svého systému zahrnout.



Obrázek 3 - Diagram znázorňující rozpoložení kol a napnutí pásu[1]

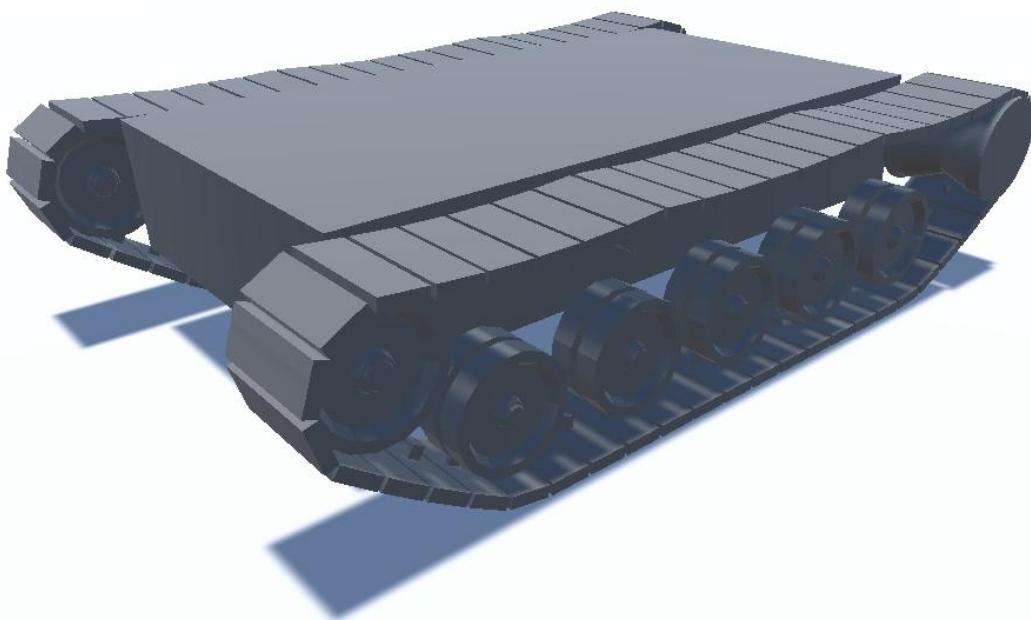
Když jsem měl tedy stanoveny všechny typy kol, mohl jsem začít řešit jejich možná spojení. Jak je evidentní z diagramu, některá kola by jednoduše nikdy neměla mít pás mezi sebou přímo natažený, např. jakékoli kolo pojazdové a kolo kladkové, nebo kolo pohonné a kolo napínací (za předpokladu přítomnosti kladkových kol). Pro správné propojení tedy stačilo definovat možné propojení sousedních kol, a následně u každého

kola najít 2 nejbližší sousední kola. Překvapivě tato část nebyla příliš náročná na naprogramování a po troše experimentování se mi povedlo vytvořit vizualizaci mezi propojenými koly.

S dokončenou detekcí a propojením správných kol zbývalo pouze generování pásu podél křivky, zde byl ovšem problém. Aby tato generace byla možná, musel jsem nejdříve vytvořit systém dovolující vytváření a úpravu křivek a následně nějakým způsobem adaptovat svůj existující generační systém na funkci generování podél křivky. Jelikož jsem s křivkami neměl žádné zkušenosti, rozhodl jsem se obrátit na internet namísto vymýšlení vlastního, komplikovaného a velmi pravděpodobně nefunkčního systému. Podařilo se mi najít příručku na sestavení křivkového systému a přes menší překážky se mi dle ní povedlo sestavit funkční editor křivek i spolu s generátorem libovolných předmětů na křivce. Vestavěný generátor měl ale několik nedostatků, které pro mě byly velice problematické. Zaprvé se nedal určit rozestup mezi jednotlivými objekty na křivce, takže by bylo třeba rozestup odhadovat, což by nebyl příliš velký problém. Problémem větším ale byly variace v rozestupech objektů, které byly ovlivňovány vzdáleností jednotlivých bodů od sebe, což by mělo za následek roztahování a smršťování pásu na různých místech. Tento problém byl pro mě velikým zklamáním, protože systém by jinak fungoval výborně. K mému nadšení se mi povedlo kontaktovat autora příručky a společně jsme dorazili k teoretickému řešení mého problému. Bohužel mi ale již nezbýl čas na jeho implementaci, a tak byl celý systém ponechán v nedodělaném, experimentálním stádiu, kde zůstává i dnes v den psaní této zprávy.

Jelikož jsem nemohl pásy na tank přidat automatizovaně, rozhodl jsem se pro hrubší řešení. Vrátil jsem se zpět ke svému jednoduchému generátoru pásu a za použití fyziky jsem pásy manuálně navléknul na tank. Toto řešení sice fungovalo, ale bylo velice časově náročné. Kdykoliv bylo pás potřeba upravit, musel být znova vygenerován, a tudíž i znova nasazen. Navíc při nasazování často docházelo k lidským chybám, které většinou vedly k potřebě restartovat celý proces nasazování. Osazování jednoho tanku tak většinou zabralo 10-15 minut, když šlo vše relativně hladce, a celkově jsem své tanky dohromady musel osazovat pásy přibližně 30x.

S navlečenými pásy nastal čas na další zajímavou část, a to samotný pohon tanku. Do této chvíle jsem se tank s pásy nepokoušel rozpohybovat, a tak na mne začala dopadat nervozita, jelikož i s prodloužením termínu jsem měl pouze pár dní na to, abych odevzdal svůj výstup v souladu s harmonogramem. Jelikož v této fázi vývoje nebyly pásy ještě plně realisticky vymodelované, nebylo možné použít k jejich pohonu ozubené kolo, jako u reálného tanku. Jako náhradu jsem tedy použil kolo normální s vysokým třením, což umožnilo pohyb tanku vpřed a vzad. Problém ale nastal, když jsem se s tankem pokusil otočit. To totiž vyžadovalo přepnutí směru jednoho z pásů, a když jsem se o toto pokusil tak se tank buďto vůbec nehnul, anebo se jeden z pásů přetrhнул. Musel jsem tedy něco udělat a nejrychlejším řešením bylo drasticky snížit tření samotných pásů se zemí. Po této úpravě bylo s tankem opravdu možné zatáčet, zároveň se ale přestával chovat jako tank kvůli sníženému tření a začalo docházet ke skluzu po zemi jako by byl tank na ledě. Čas ale nebyl nazbyt, a tak jsem se rozhodl, že tento nedostatek opravím později.

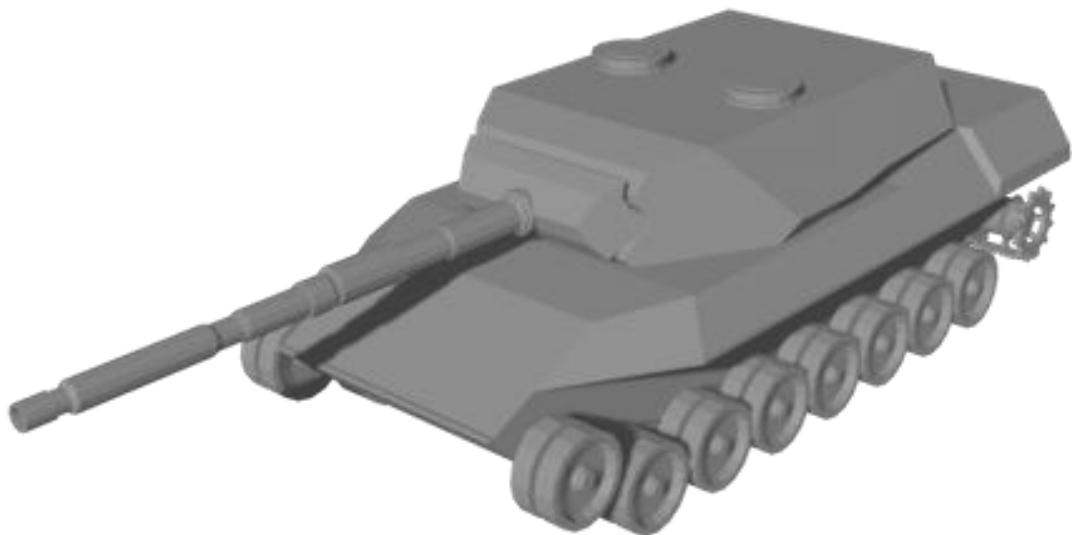


Obrázek 4 - Tank v této fázi vývoje

V momentální stavu byl tank sice pohyblivý, ale ne ovladatelny hráčem, což samozřejmě nebyl stav ideální. Čirou náhodou jsem ale byl během minulého roku vyvinul při práci na studentském projektu zodpovědný za veškerou logiku a ovládání aut, a tak jsem mohl jednoduše vzít svůj poměrně lehce upravitelný systém na přepínání mezi ovládáním hráče a vozidla a také lehce upravit svůj skript zodpovědný za samotné

řízení auta a předělat ho tak, aby lépe vyhovoval mojí potřebě řídit tank. Tato úprava byla poměrně rychlá a přibližně do hodiny jsem měl vše zprovozněno.

Můj další velký úkol bylo zprovoznění simulovaného odpružení podvozku. Jelikož jsem ale doted' nevytvořil žádný model tanku a jelikož můj momentální prototyp vzhledově zaostával a odpružení pro něj nebylo vůbec plánováno, měl jsem plán na dalších pár dní rozhodnut. Bylo tedy třeba vytvoření nového modelu a to spolu s veškerými detaily podvozku. Rozhodl jsem se pro vytvoření tanku dle Německého Leopardu 1A4 a po nalezení vhodného plánu jsem se dal do práce. Modelování probíhalo bez větších problémů a po nějaké době jsem měl hotový použitelný model tanku s podvozkem připraveným na simulování veškerého odpružení.



Obrázek 5 - Model tanku Leopard 1A4



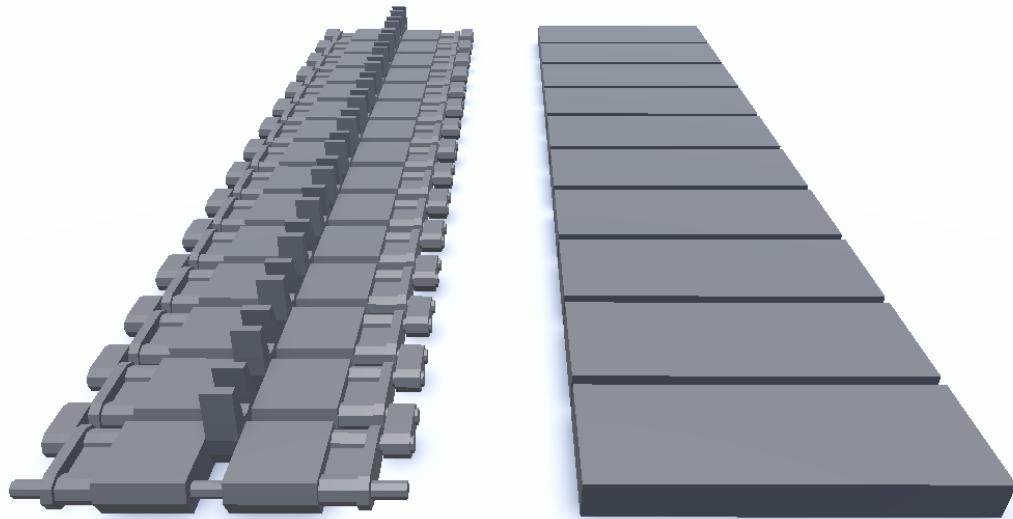
Obrázek 6 - Podvozek předešlého tanku z vnitřní strany

S vymodelovaným podvozkem jsem se mohl vrátit zpět k nastavování simulace. V momentálním stavu by totiž podvozek jednoduše odpadl od tanku, takže bylo třeba nastavit správné umístění úchytových a otočných bodů, aby podvozek byl pevně spojený s tankem. Toto nebyla příliš náročná práce, ale jednalo se opět o časově náročný úkon, jelikož každé kolo, náprava či úchytové rameno bylo třeba nastavit samostatně a nešlo využít kopírování nastavení z jedné části podvozku na druhou. Po asi hodině jsem ale měl tank plně odpružený a jediné co zbývalo bylo doladit hodnoty podvozku aby tank příliš neskákal a aby podvozek nebyl až příliš tvrdý.

Jelikož jsem při modelování podvozku udělal i ozubené hnací kolo, mohl jsem začít posouvat pásy tanku na detailnější úroveň. Přestože jsem ale měl model kola, kolize pro něj nebyly dostupné, a tak mne čekalo programování dalšího nástroje. Naštěstí jsem mohl jednoduše znova použít kód ze svého generátoru válců a lehce ho upravit a generátor ozubených kol byl na světě.

Byl opět čas na modelování, tentokrát byl třeba nový model pro články pásu. Zde se ale objevil nový problém. Sehnat plánek pásu nebylo jednoduché, protože i když je tento tank starý téměř 60 let jedná se stále o relativně moderní technologie, a tak nejsou plánky pro každou jeho část. Když se mi jeden plánek najít povedlo, přišel problém další. Po dokončení modelování jsem totiž zjistil, že nový pás vůbec nezapadá do mého ozubeného kola. Trvalo mi tedy dobrou chvíli a mnoho úprav, než jsem přišel na způsob, který mi umožnil upravit hnací kolo i článek pásu tak, aby do sebe zapadalý.

Nový článek pásu sebou nesl ale další problém. Můj jednoduchý generátor pásu totiž nebyl dostačující. Narozdíl od původního, primitivního, jednodílného článku se nový model zkládal ze čtyř samostatných částí, které bylo třeba vygenerovat na správných pozicích a správně propojit.



Obrázek 7 - Porovnání nových (levo) a starých (pravo) pásových článků

Z důvodu mnohonásobně větší komplexnosti nového článku bylo třeba napsat kompletně nový generovací systém, což bylo něco, s čímž jsem počítal, ale odkládal jsem to na co nejpozdější dobu z důvodu obav o komplikovanosti takového systému. Jak se ale ukázalo, nejednalo se o příliš těžký úkol a samotný generační kód se příliš nelišil od předchozího generátoru, akorát musel být rozšířen o mnoho řádků specifikující generaci nově přidaných částí. Změnil jsem také funkcionalitu systému. Doted' byl totiž tento skript vždy přidán na objekt, ze kterého měl být vytvořen pás. Jelikož to nyní ale nebylo možné z důvodu několika částí, musel být skript přidán na „hlavní“ objekt, pod kterým se následně vygeneroval specifikovaný pás. Když jsem poté viděl výsledný pás, byl jsem velice nadšen.

Nyní již stačilo nový pás navléknout na tank a já mohl poprvé pořádně vyzkoušet nový odpružený podvozek. Díky přidání odpružení dokázal pás vydržet mnohem větší nárazy či pády z větších výšek. Přejíždění přes překážky nyní bylo také mnohem plynulejší a díky podvozku během něj šla udržet plynulá rychlosť vozidla.



Obrázek 8 - Tank osazený novými pásy

Hlavní část projektu byla konečně dokončena, a já si mohl oddychnout, jelikož mne nyní čekalo hlavně upravování simulace a testování chování tanku. Přece jen mi ale na tanku chyběla funkční věž a dělo, a tak jsem se pustil do nápravy této chyby. Když už byl celý tank ovládán pomocí fyziky, rozhodl jsem se, že tak bude ovládána i věž a dělo. Ze všeho nejdřív jsem ale musel vytvořit systém umožňující míření. Momentálně se totiž hráč mohl během jízdy v tanku ohlížet kolem sebe ve třetí osobě, ale nebylo zcela jasné, kam přesně míří.

Rozhodl jsem se, že zaměřovač bude prozatím v podobě 3D objektu který bude zobrazen na místě, na které hráč zrovna míří svou kamerou. Abych získal tento bod, musel jsem vyslat paprsek z kamery a následně získat jeho pozici dopadu. Jelikož jsem ale nechtěl, aby tento paprsek mohl dopadnout na tank, musel jsem využít komplikovanějšího způsobu získávání pozice dopadu. Naštěstí tento způsob fungoval přesně tak, jak jsem chtěl, a já mohl na jeho pozici umístit svůj zaměřovač.

S hotovým zaměřovačem mi akorát zbývalo vyřešit otáčení děla a věže. Naštěstí jsem mohl využít stejného principu jak u věže, tak i u děla. Nejprve bylo třeba zjistit svýraný úhel mezi věží / dělem a cílovým bodem. Pokud byl úhel negativní, bylo třeba otočit věž doleva / snížit dělo dolu, naopak pokud byl úhel pozitivní bylo třeba otočit věž doprava / zvýšit dělo nahoru. Když se pak dělo či věž začaly blížit svým úhlem k nule,

snížila se rychlosť ovládania, aby daný komponent neprestrelil svôj cíl kvôli veľkej rychlosťi.

Jako posledný ze všeho bolo na řadě pridanie streľby. Chtiel jsem pre strely opäť použiť fyzikálnich objektu, ale to pribinilo komplikacie. Jelikož se strely tankov pohybujú veľkou rychlosťou, docházalo k fyzikálnej nestabiliti. Hra totiž nedokázala simulovať pohyb strely dosť detailne, a tak nebylo neobvyklé, že vystrelená munice proletela bez sebemenčeho problémou skrze zed. Musel jsem tedy využiť detekciu kolízii pri vysokých rychlosťach, kdy je pri každém výpočtu fyziky vyslan ze strely paprsek smereom na minulou poziciu, a pokud je nalezena prekážka, znamená to, že strela skrze ni proletela, a tudíž je možné simuloval její výbuch na poziciu, kde sa s prekážkou pôvodne mela setkať.

4 Závěr

Přes veškeré útrapy a překážky se mi povedlo naplnit většinu stanovených cílů a já jsem osobně s finálním produktem velmi spokojen. Byly cíle, které se mi bohužel naplnit nepovedlo, z důvodů příliš velkých ambicí a velké časové náročnosti, ale i z nich jsem získal mnoho nových zkušeností

5 Seznam zdrojů

- [1] Pingxin Wang, Xiaoting Rui, Hailong Yu. Tracked Vehicle Diagram. ScienceDirect.com [online]. 2019 [cit. 2022-03-12]. Dostupné z <https://www.sciencedirect.com/science/article/abs/pii/S0888327019304224>

6 Seznam obrázků

Obrázek 1 - Cvičný "low-poly" model	6
Obrázek 2 - Primitívny vygenerovaný pás	9
Obrázek 3 - Diagram znázorňujúci rozpoloženie kol a napnutie pásu[1]	10
Obrázek 4 - Tank v tejto fázi vývoje	12
Obrázek 5 - Model tanku Leopard 1A4	13
Obrázek 6 - Podvozok predešlého tanku z vnitriň strany	14
Obrázek 7 - Porovnanie nových (levo) a starých (pravo) pásových článkov	15
Obrázek 8 - Tank osazený novými pásy	16

7 Seznam tabulek

Tabulka 1 - Management software.....	4
Tabulka 2 - 3D engine.....	4

8 Seznam příloh

Příloha 1 - Plakat

Maturitní práce

TVORBA MATERIÁLŮ PRO BUDOUCÍ HERNÍ PROJEKTY



Autor: Jan Frídl

Vedoucí práce: Adam Přibyl

Oponent: Mgr. Marie Souchová

SMÍCHOVSKÁ STŘEDNÍ PRŮMYSLOVÁ ŠKOLA A GYMNÁZIUM 2021/2022