

image/logo-skoly.png

ZÁVĚREČNÁ STUDIJNÍ PRÁCE

dokumentace

FreeVat
Aplikace pro správu 3D modelů



Autor: Filip Podeszwa

Obor: 18-20-M/01 INFORMAČNÍ TECHNOLOGIE
se zaměřením na počítačové sítě a programování

Třída: IT4

Školní rok: 2025/26

Poděkování

Rád bych poděkoval všem umělým inteligencím, zejména **Google Gemini**, která mi pomáhala od začátku do konce, dále **ChatGPT** a taky **DeepSeek**, který není vůbec špatný. Zpočátku mi pomáhal dokonce i **GitHub Copilot**, ale ten se na mě téměř před čtvrt rokem vykašlal :(

Co se týče lidí, rád bych poděkoval panu **Ing. Petru Grussmannovi** za pomoc v Kubernetesu, která se ukázala být k ničemu, ale to je můj problém, protože jsem aplikaci již nestihl zdockerovat. Nasazení aplikace na web se budu věnovat někdy později.

Prohlášení

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým a prezentačním účelům na Střední průmyslové a umělecké škole v Opavě, Praskova 399/8.

V Opavě 7. 1. 2026

.....
Podpis autora

Abstrakt

FreeVat je webová platforma zaměřená na komunitní **sdílení, prohlížení a správu** 3D modelů. Hlavním cílem aplikace je poskytnout uživatelům intuitivní prostředí pro prezentaci jejich digitální tvorby **bez nutnosti instalace** specializovaného softwaru. Platforma slouží jako centrální úložiště, které propojuje tvůrce 3D obsahu s koncovými uživateli, kteří tyto modely mohou využít ve **vlastních projektech, hrách nebo při 3D tisku**.

Aplikace umožňuje uživatelům **nahrávat modely, zobrazovat je ve 360°, spravovat je a taky mazat**. Přihlašování je možné jak klasickou metodou jméno/email/heslo, tak pomocí **Google** či **GitHubu**. Poskytuje taky prostředí pro psaní komentářů na ostatní modely a zobrazuje základní informace o nich.

Klíčová slova

FreeVat, 3D grafika, 3D modely, Django, Python, Three.js, HTML, CSS, JavaScript, Tailwind, Adobe Photoshop, PostgreSQL, Node.js, open-source, správa digitálního obsahu, webová aplikace, interaktivní prohlížeč, web, webové stránky, digitální grafika, programování, modelování

Abstract

FreeVat is a web platform focused on community **sharing, viewing, and managing** 3D models. The main goal of the application is to provide users with an intuitive environment for presenting their digital creations **without the need to install** specialized software. The platform serves as a central repository that connects 3D content creators with end users who can use these models in **their own projects, games, or 3D printing**.

The application allows users to **upload models, view them in 360°, manage them, and delete them**. Users can log in using the classic name/email/password method, but it is also possible via **Google** or **GitHub**. It also provides an environment for commenting on other models and displays basic information about them.

Keywords

FreeVat, 3D graphics, 3D models, Django, Python, Three.js, HTML, CSS, JavaScript, Tailwind, Adobe Photoshop, PostgreSQL, Node.js, open-source, digital content management, web app, interactive viewer, web, web pages, digital graphics, programming, modeling

Obsah

Úvod	1
1 Využité technologie	2
1.1 Front-end	2
1.2 Back-end	4
2 Vývoj	8
2.1 Indev 0.0.1	9
2.2 Indev 0.0.5	10
2.3 Indev 0.0.7	11
2.4 Indev 0.0.9	12
2.5 Pre-Alpha 0.1.0	13
2.6 Alpha 1.0.0	15
3 Splněné a nesplněné cíle	17
3.1 Splněno	17
3.2 Nesplněno/Neimplementováno	17
Závěr	18

ÚVOD

Dřív, než se ponoříme do samotného popisu vývoje aplikace, chtěl bych uvést na pravou míru, že toto byl můj první "velký" projekt, co se IT týče. Nikdy předtím jsem nic takového nedělal. Toto bylo poprvé, co jsem musel pracovat s více věcmi najednou. Nestačilo např. jen něco vytvořit v jazyce *C* či *HTML*, musel jsem do sebe spojit více technologií a zajistit, aby spolu řádně komunikovaly.

Proto prosím berte na vědomí, že postupy v mé aplikaci mnohdy nemusí zcela vyhovovat zažitým praktikám, občas se i vyskytuje malinké množství duplicitního kódu, který samozřejmě v žádné správně naprogramované aplikaci není úplně žádoucí. Stejně tak nemám do hloubky prostudované frameworky, které jsem k vývoji použil. Je tudíž možné, že udělám nějaký překlep nebo i něco špatně interpretuji.

1 VYUŽITÉ TECHNOLOGIE

Tato kapitola obsahuje podrobný popis technologií, které byly k projektu **FreeVat** využity. Jsou to jak technologie, co se starají o vzhled, tak i ty, co jsou zodpovědné za vnitřní logiku i běh aplikace. Všechny zdroje jsou uvedeny na konci dokumentace v sekci **Zdroje**.

1.1 FRONT-END

V této sekci jsou popsány technologie využité při vývoji front-endové části aplikace, která tvoří rozhraní pro interakci s uživatelem. Základním stavebním kamenem je kombinace jazyků **HTML, CSS a JavaScriptu**, doplněná o framework **Tailwind CSS** pro efektivní tvorbu moderního a responzivního designu. Klíčovou roli zde hraje knihovna **Three.js**, která do prostředí webového prohlížeče integruje pokročilou 3D grafiku, zatímco o vizuální prvky a grafické podklady se postaral software **Adobe Photoshop**. Cílem této části je objasnit výběr těchto nástrojů a jejich přínos pro výslednou vizuální podobu a funkčnost uživatelského prostředí.

1.1.1 Tailwind CSS

Při stylování stránek jsem se rozhodl využít framework Tailwind CSS. Je lepší než klasické CSS, protože využívá tzv. **utility-first**, což jsou předpřipravené třídy, které lze spojovat a ušetřit tak veliké množství kódu.

```
.example {  
    text-align: center;  
    padding: 0.5rem;  
    display: flex;  
}
```

Klasické CSS styly

```
.example {  
    @apply text-center;  
    @apply p-2;  
    @apply flex;  
}
```

Tailwind "utility-first"

1.1.2 Three.js

Klíčovou roli v aplikaci hraje Three.js. Je to populární open-source knihovna, která výrazně zjednodušuje tvorbu interaktivní 3D grafiky přímo v prohlížeči. Funguje jako nadstavba nad rozhraním WebGL, což umožňuje plně využít výkon grafické karty pro hardwarově akcelerované renderování v reálném čase.

```
/* Inicializace WebGL rendereru s vyhlažováním a průhledným pozadím */
const renderer : WebGLRenderer = new THREE.WebGLRenderer( {antialias: true, alpha: true});

renderer.setSize(container.clientWidth, container.clientHeight);
renderer.setPixelRatio(window.devicePixelRatio);
/* Nastavení mapování tónů a barevného prostoru pro realistický vzhled */
renderer.toneMapping = THREE.ACESFilmicToneMapping;
renderer.outputColorSpace = THREE.SRGBColorSpace;

/* Vyčištění kontejneru a vložení plátna (canvas) prohlížeče */
container.innerHTML = '';
container.appendChild(renderer.domElement);

/* Ambientní světlo pro základní prosvětlení stínů */
scene.add(new THREE.AmbientLight( {color: 0xffffffff, intensity: 1.5}));
```

Ukázka kódu s použitím Three.js

1.1.3 Adobe Photoshop 2026

Ačkoliv je Photoshop primárně rastrový editor, je možné v něm tvořit i nějaké to jednoduché **SVG**. V tomto programu byly vytvořeny **vlajky a logo aplikace**.

1.2 BACK-END

Tato sekce se obsahuje popis technologií při návrhu a implementaci back-endové části aplikace, která zajišťuje logiku systému a správu dat. Hlavním pilířem je framework Django - robustní základ pro serverovou část, doplněný o nástroje Crispy Forms pro pokročilé zpracování formulářů a Rosetta pro zajištění vícejazyčného rozhraní. Pro bezpečné a výkonné ukládání informací byl zvolen databázový systém PostgreSQL, zatímco integraci externích služeb a bezpečné přihlašování zajišťuje protokol OAuth. Celý vývojový proces a efektivní sestavení aplikace (buildování) pak doplňuje prostředí Node.js spolu s moderním nástrojem Vite.

1.2.1 Django

Jako základ projektu jsem zvolil framework Django, postavený na čitelném jazyku Python. Díky přístupu „batteries-included“ a integrovanému ORM výrazně urychluje vývoj a zajišťuje vysokou úroveň bezpečnosti proti běžným útokům. Architektura je navržena pro snadnou škálovatelnost a podporuje čistý, pragmatický design celého systému.



1.2.2 Python knihovny



Rosetta

Protože jsem chtěl udělat web ve 3 různých jazycích, potřeboval jsem efektivně vytvářet a spravovat překlady, a na to jsem využil knihovnu Rosetta. Díky této knihovně je možné spravovat překlady přímo v admin panelu jednoduše a přehledně.

Crispy Forms

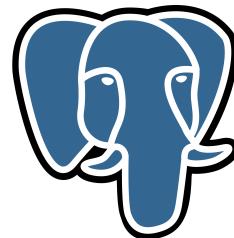
Tento balík jsem využil na tvorbu a stylování uživatelských formulářů. Návrh tak šel mnohem rychleji, než kdybych je měl programovat od páky. Za zmínu taky stojí, že pomocí malé knihovničky je možné Crispy Forms propojit s Tailwindem a je možné si tak vytvářet vlastní stylované formuláře, než používat ošklivý vestavěný design.

PyYAML

Díky této knihovně jsem databázové dumpy mohl uložit do formátu **YAML**, který je pro člověka mnohem lépe čitelný než obyčejný **JSON** a za mě je i spolehlivější.

1.2.3 PostgreSQL

Django sice má v sobě vestavěnou databázi **SQLite**, ale ta je hodně omezená a pro větší projekty nevhodná. Proto jsem sáhnul po robustnějším řešení a nainstaloval **PostgreSQL**, což je velice výkonný **objektově-relační** databázový systém. Je známý svou spolehlivostí a integritou dat. A hlavně je open-source, takže zadarmo!



1.2.4 OAuth

OAuth je otevřený protokol, který v aplikaci zajišťuje bezpečné ověřování uživatelů prostřednictvím třetích stran, jako je Google, Facebook nebo GitHub. Díky tomu se uživatelé nemusí starat o hesla a jednoduše se kdykoli přihlásí a odhlásí jedním kliknutím.



Přihlásit se k FreeVatu

Ještě nejste členem? [Vytvořit nový účet!](#)

Email *

Zadejte svůj email

Heslo *

Zadejte své heslo

Zapamatovat si mě [Zapomenuté heslo?](#)

Přihlásit se k účtu

nebo

[Google](#) [GitHub](#)

Přihlašovací formulář

1.2.5 Node.js a Vite

Pro integraci technologií **Tailwind CSS** a **Three.js** bylo nezbytné zvolit moderní vývojové prostředí schopné efektivně kompilovat zdrojové kódy a styly. Klíčovou roli zde hraje **Node.js**, který poskytuje runtime prostředí pro buildovací procesy. Nástroj **Vite** zajišťuje extrémně rychlou obsluhu kódu a jeho optimalizaci pro produkční nasazení. Díky technologii Hot Module Replacement (HMR) umožňuje Vite promítat změny v JavaScriptu a CSS okamžitě do prohlížeče, což zásadně zrychluje a zpříjemňuje celý proces vývoje aplikace.



2 VÝVOJ

V této kapitole jsou podrobně rozepsány všechny verze projektu. V každé je uvedeno, co bylo přidáno, co odebráno nebo co opraveno.

2.1 INDEV 0.0.1

V červnu létě 2025 jsem udělal ten úplně první krok - vytvořil jsem repozitář na GitHubu. Hned nato jsem přidal soubor **README.md**, do kterého jsem vypsal všechny informace o projektu - popis, závislosti, cíle...

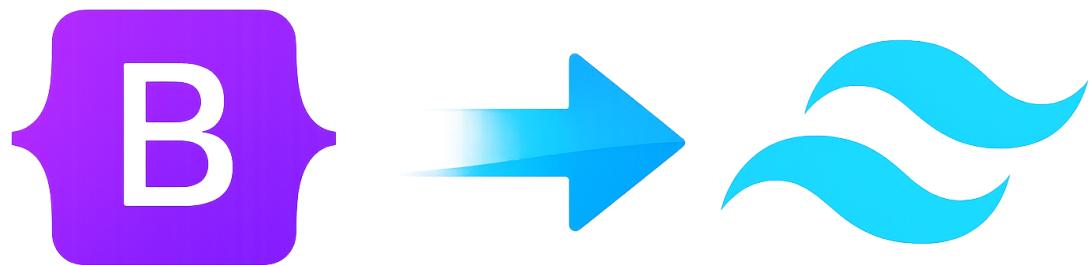
The screenshot shows the GitHub README page for the 'FreeVat' project. At the top, there's a title section with a gift icon and the text 'FreeVat'. Below it is a detailed description of the application: 'FreeVat je webová aplikace, která umožňuje uživatelům nahrávat, zobrazovat, mazat a ukládat 3D modely.' It highlights that the app is built on modern technologies and provides a simple and intuitive interface for viewing 3D models. Below the description are three status badges: 'Verze Alpha 1.0.0', 'Vývojář PanVát', and 'Licence Open Source'. Under the 'Technologie' heading, there's a 'Front-end' section listing the following technologies: HTML, CSS & JS, Three.js, Tailwind CSS, and Adobe Photoshop.

- `HTML, CSS & JS` - Základní struktura, stylování a logika
- `Three.js` - Práce s 3D grafikou na webu
- `Tailwind CSS` - Moderní a responzivní webový design
- `Adobe Photoshop` - Tvorba grafických prvků

Vzápětí jsem zapnul **PyCharm**, zprovoznil **virtuální prostředí** a nainstaloval Django aplikaci. Během toho jsem přemýšlel i nad jménem. Napadlo mě více možností, ale vyhrál **FreeVat**. Nakonec jsem nainstaloval pár nezbytných závislostí.

2.2 INDEV 0.0.5

Původně jsem místo **Tailwindu** používal **Bootstrap**, ale později jsem zjistil, že je omezenější a není tak flexibilní jako **Tailwind**.

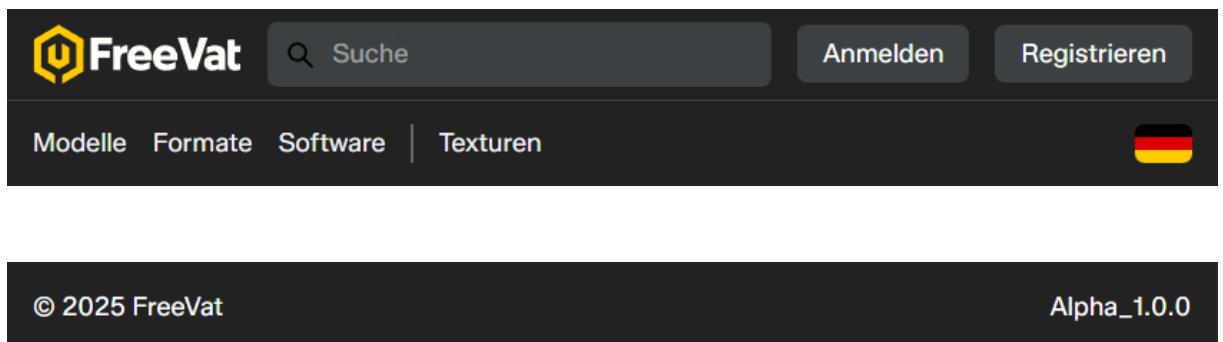


Tento dobou už jsem pracoval na záhlaví, navigačním menu a zápatí webu. Dokonce se mi podařilo zprovoznit tlačítko pro přepínání jazyků, a tak už tehdy byl web v Češtině, Angličtině a Němčině. Přepínání je zprovozněno pomocí JavaScriptu.



2.3 INDEV 0.0.7

V říjnu jsem dokončil záhlaví, navigaci a zápatí. Přidal jsem dokonce vyhledávací lištu (která ovšem zatím není funkční :)

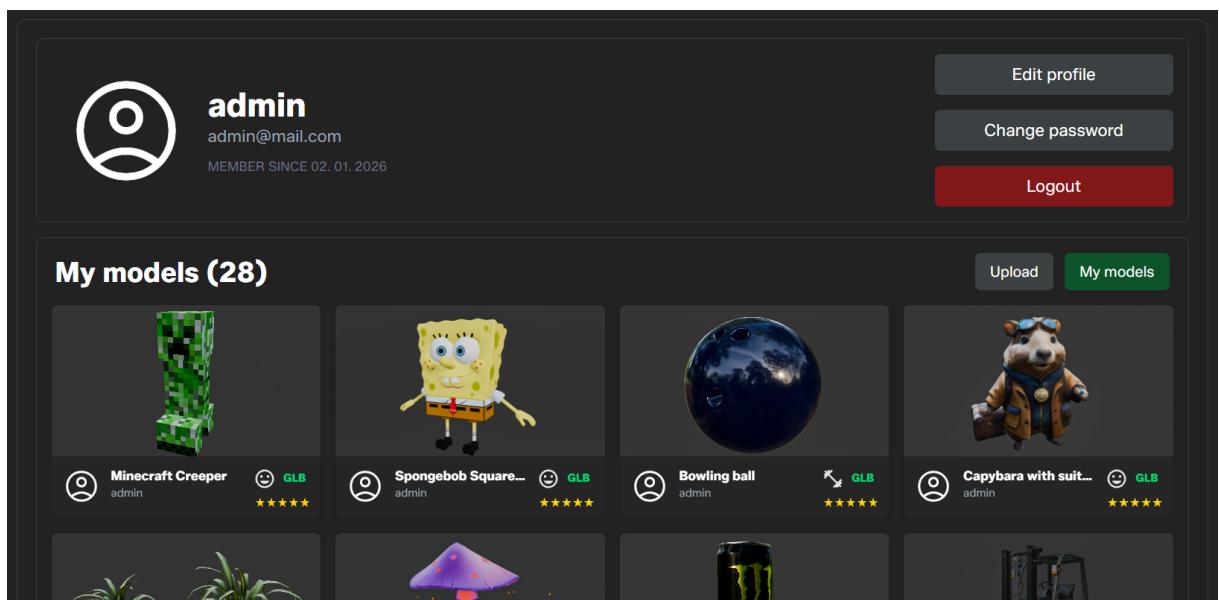


Dokonce v té době proběhl první pokus o implementaci 3D prohlížeče na web a dokonce se to i podařilo, ačkoliv to nefungovalo tak, jak by mělo.

```
[0] > freevat@1.0.0 tailwind:dev
[0] > npx @tailwindcss/cli -i ./static/css/tailwind/src/styles.css -o ./static/css/tailwind/dist/styles.css --watch
[0]
[1]
[1]  VITE v7.3.0 ready in 503 ms
[1]
[1]  → Local:  http://localhost:5173/
[1]  → Network: use --host to expose
[0] ≈ tailwindcss v4.1.17
[0]
[0] Done in 229ms
```

2.4 INDEV 0.0.9

Zprovoznění formulářů pro registraci nových účtů a přihlašování se ke stávajícím účtům. Fun-
guje jak klasické přihlášení, tak i OAuth přes Google a GitHub.



2.5 PRE-ALPHA 0.1.0

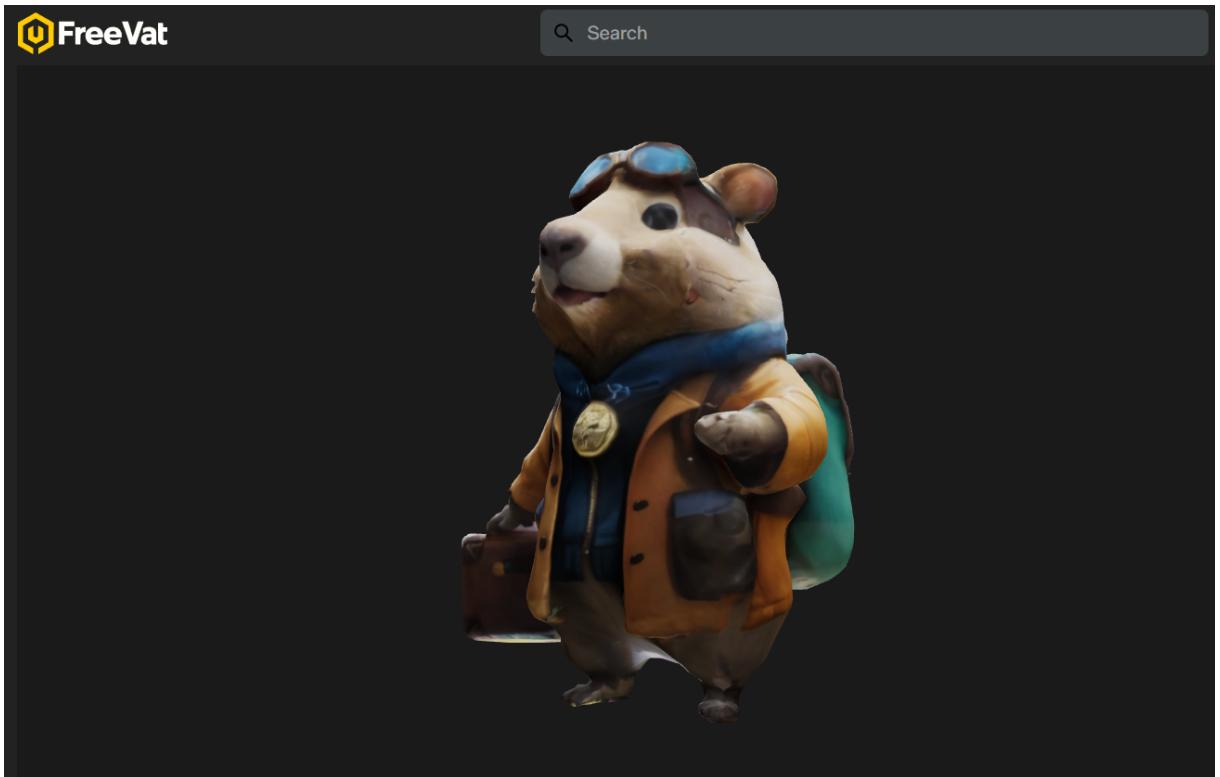
Tohle byl docela velký update, protože vyšla první Pre-Alpha. Podařilo se mi zprovoznit formulář pro nahrávání 3D modelů, které se nyní krásně ukládají do databáze i se všemi daty.

The screenshot shows a user interface for uploading 3D models. It consists of three main sections:

- 3D Model ***: A section for uploading the main 3D model file. It includes a placeholder area labeled "Upload 3D Model File", a note about the maximum file size (100 MB for .fbx, .gltf, .blend files), and a "Select" button.
- Thumbnail ***: A section for selecting a thumbnail image. It includes a placeholder area labeled "Main Preview Image", a note about the maximum file size (2 MB for .jpg, .png files), and a "Select" button.
- Photos**: A section for adding additional photos. It includes a placeholder area labeled "Photos", a note encouraging users to upload more images to show details of their model, and two buttons: "Select" and "Delete".

At the bottom of the form are three large buttons: "Upload" (green), "Reset" (grey), and "Discard" (red).

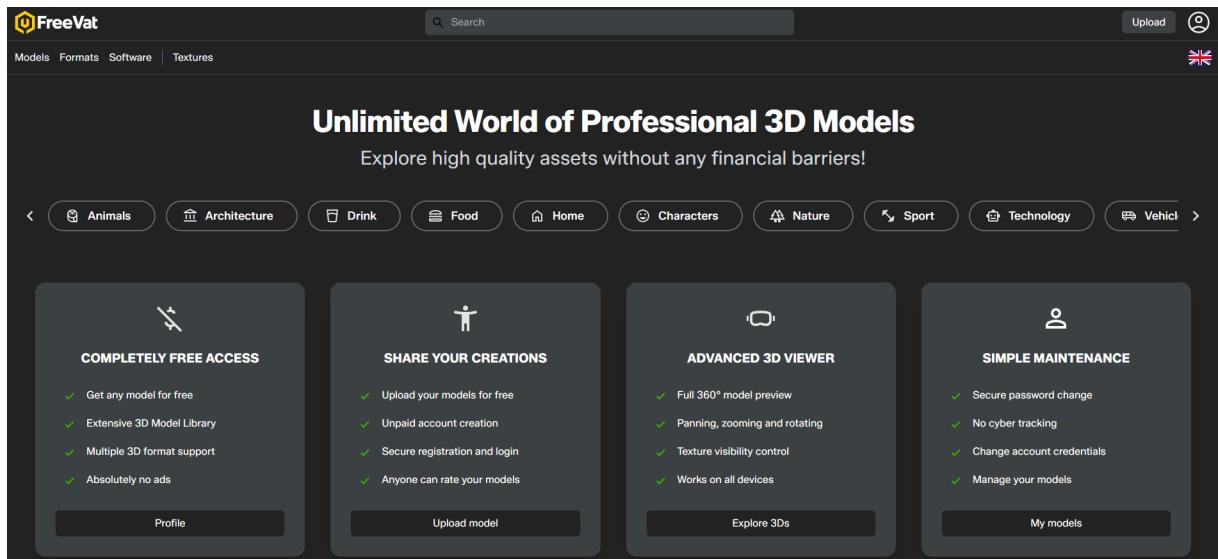
Taky 3D viewer se mi podařilo plně zprovoznit. Nyní je možné model posouvat, přiblížovat a rotovat.



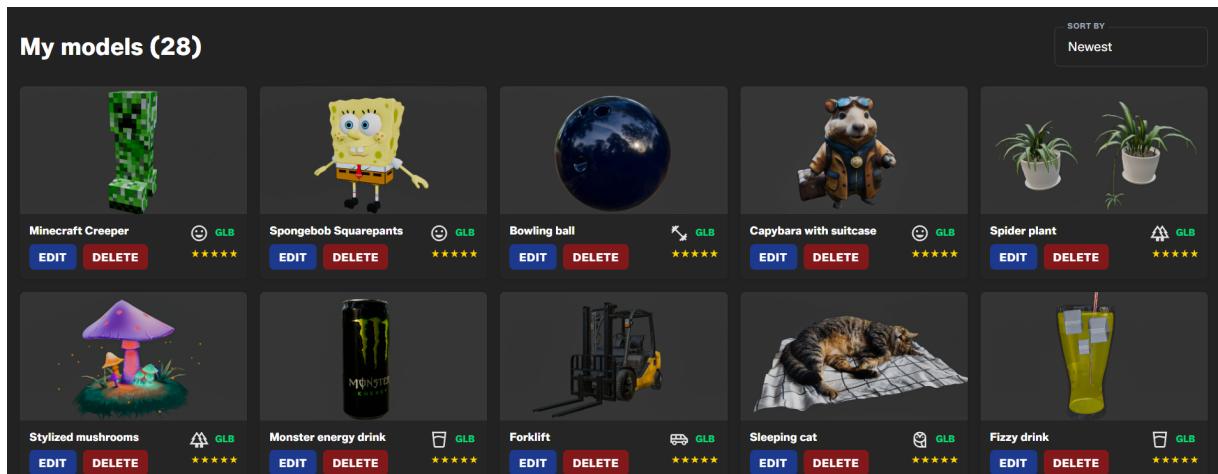
2.6 ALPHA 1.0.0

A je to tady! Vyšla historicky první Alpha verze mého projektu. V této fázi je připravená k používání - soubory se správně nahrávají, 3D zobrazení funguje, uživatelé můžou stahovat modely a psát komentáře

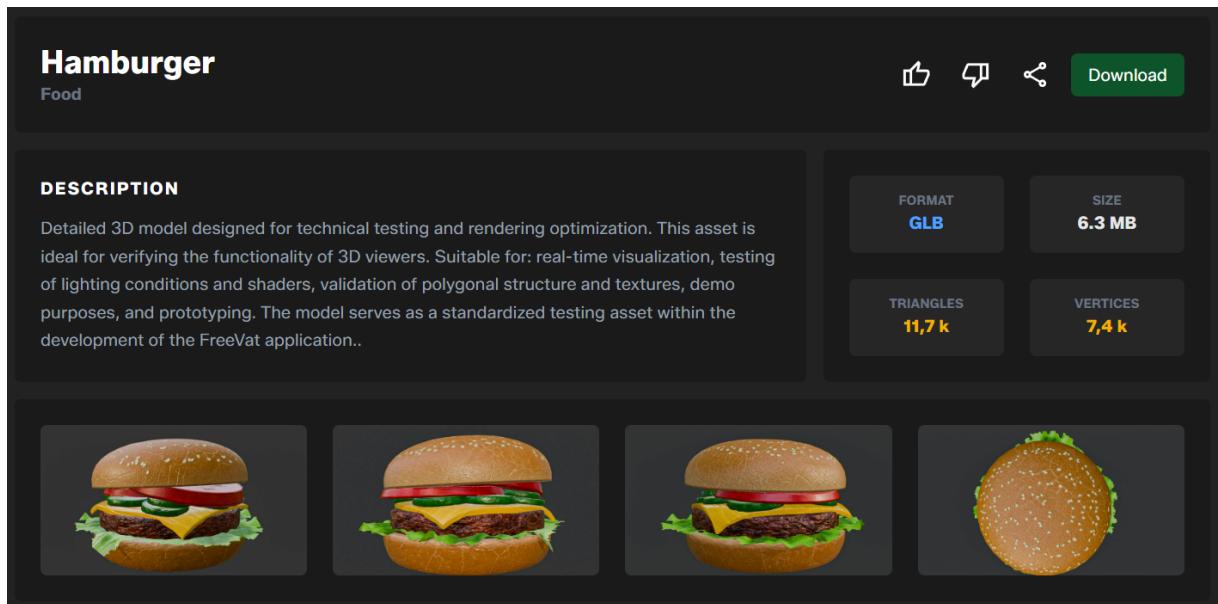
Domovská obrazovka je hezky nadesignovaná



Uživatelé mohou upravovat nebo mazat své nahrané modely



O modelu se zobrazují technické informace, jako je počet polygonů a vrcholů, velikost souboru a formát modelu. Zobrazují se i čtyři náhledové obrázky.



3 SPLNĚNÉ A NESPLNĚNÉ CÍLE

3.1 SPLNĚNO

- ukládání modelů do databáze i s jejich daty
- provádění základních CRUD operací s modely
- prohlížení modelů ve 360°
- posouvání, přibližování a otáčení modelů
- registrace, přihlašování a mazání uživatelských účtů
- zobrazení základních informací o modelech
- vytvoření vícejazyčné webové aplikace
- řazení modelů podle názvu, data nahrání a velikosti

3.2 NESPLNĚNO/NEIMPLEMENTOVÁNO

- podpora formátů 3D softwarů
- plně responzivní design na PC, tabletu i mobilu
- vyhledávání 3D modelů podle názvu
- zdockerování aplikace
- a samozřejmě mnoho dalšího

ZÁVĚR

Hlavním cílem tohoto projektu bylo vytvořit komplexní webovou aplikaci schopnou ukládat a interaktivně zobrazovat 3D modely různých formátů. Realizace tohoto záměru mi umožnila nejen dosáhnout vytyčených technických cílů, ale především získat hluboké praktické porozumění frameworku **Django** a osvojit si moderní přístupy k tvorbě uživatelského rozhraní pomocí **Tailwind CSS**, kterému jsem díky tomuto projektu porozuměl do hloubky.

Jelikož se jednalo o můj první velký projekt tohoto rozsahu, musel jsem čelit mnoha výzvám, od návrhu databázové struktury až po integraci 3D grafiky v reálném čase. Celý proces vývoje mě naučil efektivně řešit komplexní problémy a propojovat různorodé technologie do jednoho stabilního celku. Aplikace FreeVat je tak výsledkem dlouhodobého učení a představuje funkční základ, který jsem připraven v budoucnu dále rozšiřovat o nové moduly a optimalizace.

LITERATURA

- [1] W3C. *HTML Living Standard*. [online]. Dostupné z: <https://html.spec.whatwg.org/> [cit. 2025-01-05].
- [2] W3C. *CSS Specifications*. [online]. Dostupné z: <https://www.w3.org/Style/CSS/Overview.en.html> [cit. 2025-01-05].
- [3] Mozilla Developer Network. *JavaScript Guide*. [online]. Dostupné z: <https://developer.mozilla.org/en-US/docs/Web/JavaScript> [cit. 2025-01-06].
- [4] Three.js Authors. *Three.js Documentation*. [online]. Dostupné z: <https://threejs.org/docs/> [cit. 2025-01-06].
- [5] Tailwind Labs. *Tailwind CSS Documentation*. [online]. Dostupné z: <https://tailwindcss.com/docs> [cit. 2025-01-07].
- [6] Adobe. *Adobe Photoshop Documentation*. [online]. Dostupné z: <https://helpx.adobe.com/photoshop/user-guide.html> [cit. 2025-01-07].
- [7] Django Software Foundation. *Django Documentation*. [online]. Dostupné z: <https://docs.djangoproject.com/> [cit. 2025-01-08].
- [8] Django Software Foundation. *Internationalization and Localization*. [online]. Dostupné z: <https://docs.djangoproject.com/en/stable/topics/i18n/> [cit. 2025-01-08].
- [9] Django Crispy Forms. *Crispy Forms Documentation*. [online]. Dostupné z: <https://django-crispy-forms.readthedocs.io/> [cit. 2025-01-09].
- [10] PostgreSQL Global Development Group. *PostgreSQL Documentation*. [online]. Dostupné z: <https://www.postgresql.org/docs/> [cit. 2025-01-09].
- [11] IETF. *The OAuth 2.0 Authorization Framework*. [online]. Dostupné z: <https://datatracker.ietf.org/doc/html/rfc6749> [cit. 2025-01-10].
- [12] OpenJS Foundation. *Node.js Documentation*. [online]. Dostupné z: <https://nodejs.org/en/docs> [cit. 2025-01-10].
- [13] Vite Team. *Vite Documentation*. [online]. Dostupné z: <https://vitejs.dev/guide/> [cit. 2025-01-11].