

**SVEUČILIŠTE U ZAGREBU**  
**GRAFIČKI FAKULTET ZAGREB**

**ZAVRŠNI RAD**

Josip Pokrajac



Sveučilište u Zagrebu  
Grafički fakultet

Smjer: Dizajn grafičkih proizvoda

# **ZAVRŠNI RAD**

## **PRIMJENA NOVIH MOGUĆNOSTI SIMULACIJE FLUIDA, VATRE I DIMA U PROGRAMU BLENDER**

Ime i prezime mentora:

prof.dr.sc. Lidija Mandić

Ime i prezime studenta:

Josip Pokrajac

Zagreb, 2021

## **SAŽETAK**

Ovaj završni rad se bavi temom simulacije fluida, vatre i dima u inačici 2.93 programa Blender. U Blender-u je moguće napraviti sve od trodimenzionalnih modela, tekstura, obrade slika, animacija, renderiranja te u novije vrijeme raznih fizikalnih simulacija. U završnom radu će se detaljno proći kroz proces izrade scene u kojoj će biti smještena fizikalna simulacija fluida i vatre te proces izrade samih simulacija. Izrada scene će obuhvatiti 3D modeliranje objekta i podloge te obradu tekstura kako bi se dobio realniji i estetski ugodniji prikaz. Za izradu simulacija proći će se kroz njihovo smještanje u prostor i postavke kojima bi se dobio realni prikaz. Na kraju se provodi renderiranje gdje će se sve napravljeno stopiti u animaciju.

# SADRŽAJ

1. UVOD .....	1
2. TEORIJSKA PRIPREMA ZA IZRADU SCENE - UVOD U BLENDER.....	2
3. IZRADA OKRUŽENJA I SCENE .....	9
3.1 Izrada piramide .....	9
3.2 Izrada pješčane podloge.....	13
3.3 Izrada zidina.....	15
3.4 Izrada .....	15
4. IZRADA SIMULACIJE FLUIDA .....	17
5. IZRADA SIMULACIJE DIMA I VATRE.....	26
6. ZAKLJUČAK .....	30

# 1. UVOD

Blender je software i sučelje za izradu računalne ili digitalne grafike kroz visokokvalitetne slike ili animacije pomoću trodimenzionalne geometrije. Blender je, za razliku od ostalih sličnih software-a, potpuno besplatan i open source, što znači da upravo ta zajednica ljudi koji koriste Blender rade na njegovom usavršavanju te naravno ljudi iz organizacije Blender Foundation koji odlučuju o glavnom smjeru razvoja Blender-a. Sve je počelo u obliku alata koji je za svoje potrebe produciranja koristila nizozemska animacijska kompanija NeoGeo, koji je osnovao začetnik, koji je i danas glavni programer, Ton Roosendaal. Pred kraj 90-ih godina prošlog stoljeća kompanija NeoGeo je omogućila download kopija Blender-a sa svojih stranica te je interes za program veličine manje od 2 MB počeo rasti. Ton Roosendaal je 1998 godine osnovao kompaniju Not a Number s ciljem marketinga i prodaje software-a Blender.

Do kraja 2000. godine Blender je brojao oko 250 tisuća korisnika širom svijeta, ali unatoč tome investitori nisu bili zadovoljni te je kompanija NaN morala zatvoriti svoja vrata. Ton je ipak našao način kako da spasi Blender, pošto je postojao veliki interes zajednice korisnika za software, osnovana je neprofitna organizacija Blender Foundation koja je sa €100,000 skupljenih od zajednice otkupila prava na Blender. Ovdje počinje razvoj besplatnog, open source programa kakvog ga danas znamo.

Interes za 3D modeliranje kod mene se razvio prije dosta godina kada sam se upoznao sa *Blender*-om prilikom izrade raznih modela mačeva, odjeće i pokušaja izrade cijelih modela životinja te animacije istih koji su služili kao prijevozno sredstvo likova jedne video igre. Odabir teme vezane uz 3D modeliranje je bio način da obnovim stara znanja i naučim nešto novo s obzirom da to područje planiram usavršiti i možda se baviti time u budućnosti.

Kranji rezultat završnog rada će pokušati na što jasniji i realniji način prikazati simulaciju kretanja fluida, u ovom slučaju vode, te plinova vatre i dima kroz prostor u kojemu se nalaze. Od verzije *Blender*-a 2.82 stari način simuliranja je potpuno zamijenjen novim simulacijskim okvirom *Mantaflow* koji je utemeljen na fizikalnim zakonima i prenosi ih u virtualni svijet. Okruženje u kojemu se scena nalazi sastoji se od piramide okružene pustinjским pijeskom, kamenjem, kanalom i obeliskom. Izvor vode je postavljen u unutrašnjem dijelu piramide u izrezima pri vrhu od kuda će se slijevati po stranicama piramide i ispuniti prostor koji omeđuje kamenje oko nje. Obelisk je objekt u koji se voda sudara i zaobilazi ga nakon sudara te ispunjava kanal na kraju kojega će biti odvod vode iz scene. Vatra i dim su simulirani u obliku baklji koje su postavljene okolo kanala, dim se sudara sa preprekama.

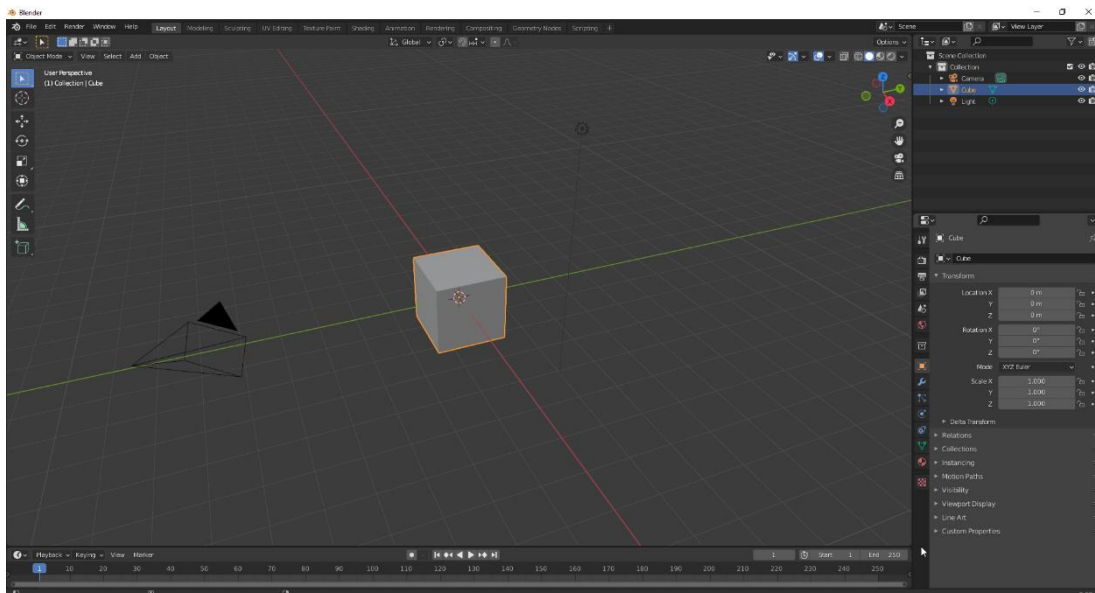
## 2. TEORIJSKA PRIPREMA ZA IZRADU SCENE - UVOD U BLENDER

*Blender* se zasniva na svojoj ideji potpunog besplatnog software-a koji svatko može koristiti i izdaje se pod *GNU General Public License* licencom, što onima koji ga koriste omogućava:

- Korištenje *Blender*-a u bilo koje svrhe
- Distribuciju *Blender*-a
- Proučavanje programskog koda na koji *Blender* radi i mijenjanje istoga
- Distribuciju izmijenjenih to jest neslužbenih verzija programa

Ovo omogućava prodaju i distribuciju svih kreacija koje netko napravi u *Blender*-u, one su isključivo vlasništvo kreatora i ne postoji mogućnost tužbi ili odricanja dijela zarade kod prodaje svojih kreacija. Kod ostalih sličnih programa može lako doći do nezakonitog korištenja pošto su ti programi izrazito skupi i većina ljudi koji se ne bave profesionalno ovom industrijskom granom dugi niz godina uglavnom koriste nelicencirane kopije što je još jedna ogromna prednost *Blender*-a čime daje jednu sigurnost svima koji žele probati ili se tek počinju profesionalno baviti 3D modeliranjem.

Svaki *Blender* projekt počinje praznim trodimenzionalnim prostorom i sivom kockom u sredini koja je na neki način postala zaštitni znak *Blender*-a. Prilikom otvaranja se svaki puta otvori *Splash screen* iz kojega je moguće otvoriti nedavne projekte ili započeti novi.



*Slika 1 Početno sučelje Blender-a*

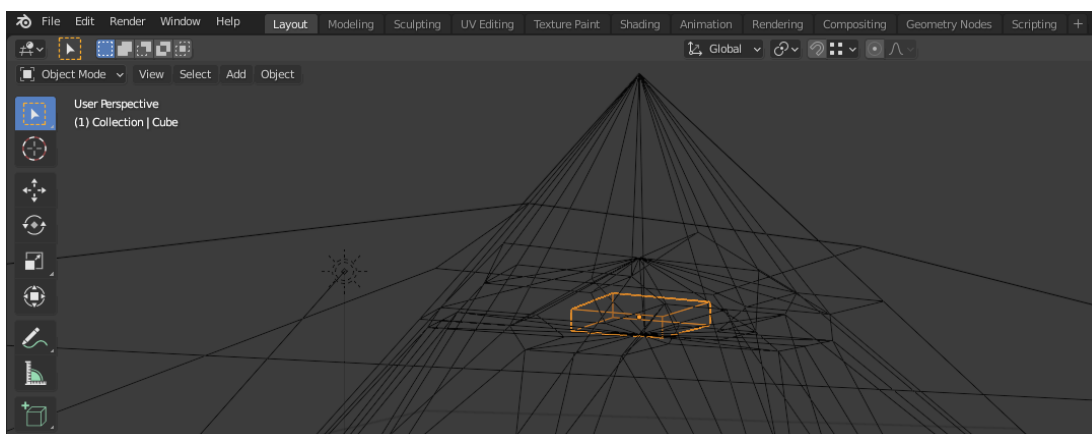
Iz početnog sučelja imamo skoro sve u vidokrugu što nam je potrebno za početak rada. U gornjem-lijevom dijelu ekrana imamo menije: *File* (služi za otvaranje, spremanje, započinjanje novih projekata), *Edit* (komande undo, redo, povijest naredbi te *Preferences* ili postavke *Blender-a* što je jako bitan dio jer u njima podešavamo sve i palimo *Add on-ove* koji su neophodni za napredniji rad), *Window* (podešavanje prozora i *workspace-a*) te *Help*. Na desno od menija *Help* imamo različite *workspace-ove* koji su jako povezani jedan s drugim:

- *Layout* – *Object mode*, način rada koji je fokusiran na pomicanje modela i olakšava snalaženje u sceni
- *Modeling* – *Edit mode*, način rada u kojemu radimo na jednom modelu, na njegovoj geometriji, bez interakcije sa drugim modelima
- *Sculpting* – način rada u kojemu oblikujemo geometriju modela kao glinu, potrebno je jako povećati rezoluciju modela da bi se moglo raditi u ovom načinu
- *UV editing* – način rada u kojemu trodimenzionalni model prenosimo u dvodimenzionalni što je priprema za teksturiranje
- *Texture paint* – način rada u kojemu je moguće ručno izraditi ili doraditi teksture
- *Shading* – način rada za izradu materijala modela, simuliranje materijala poput stakla, kamena, drva, itd...
- *Animation* – način rada za izradu i uređivanje animacija

- Rendering – način rada u kojemu dobivamo finalni proizvod, *render*, te njegove postavke
- Compositing
- Geometry nodes
- Scripting

Ispod izbornika gdje piše *Object mode* imamo alate za manipulaciju cijelim modelom u prostoru: *Move*, *Scale* i *Rotate*. *Edit mode* omogućuje uz ove još više alata pošto u njemu direktno radimo na modelu. Desno od izbornika gdje piše *Object mode* imamo izbornike:

- View - služi za podešavanje izbornika koje vidimo u sučelju i kontrolu onoga što vidimo
- Select - kontrole za označavanje objekata i dijelova modela u *edit mode*-u
- Add - služi za dodavanje osnovnih objekata u scenu
- Object - služi za detaljne postavke samih objekata ili modela



*Slika 2 Dio sučelja objašnjenog u tekstu iznad slike*

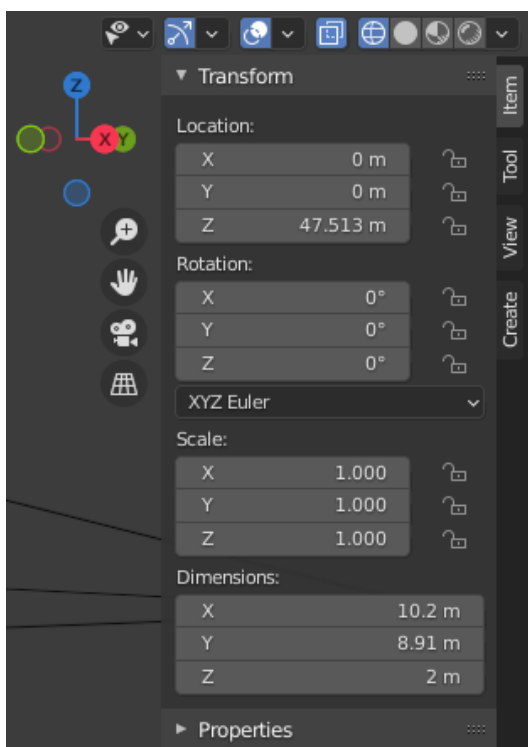
Sa desne strane prethodno opisanog sučelja nalazi se par bitnih alata. Od gornje alatne trake važno je izdvojiti četiri kruga sa desne strane što se zove *Viewport Shading*. Oni određuju na koji način će scena biti prikazana korisniku. Prvi krug, *Wireframe mode*, uključuje način u kojemu su svi objekti prozirni i vidljive su samo linije modela. Drugi krug, *Solid mode*, način prikaza gdje objekti nisu prozirni i dodana im je vanjska površina uz vidljive linije strukture. Treći krug, *Material Preview mode*, način prikaza u kojemu je objektima dodana tekstura. Četvrti krug, *Render mode*, način prikaza u



kojemu u stvarnom vremenu vidimo prikaz konačnog proizvoda sa dodanim svjetlom, sjenama, simulacijama, itd...Kod scena i sa samo nekoliko objekata prijelaz u ovaj način prikaza može lako srušiti *Blender* ako za rad ne koristimo *high end* računalo te to treba imati u vidu kako ne bi došlo do gubitka podataka.

Ispod te alatne trake s lijeve strane imamo prikaz orijentacije u *Blender*-ovom 3D prostoru.

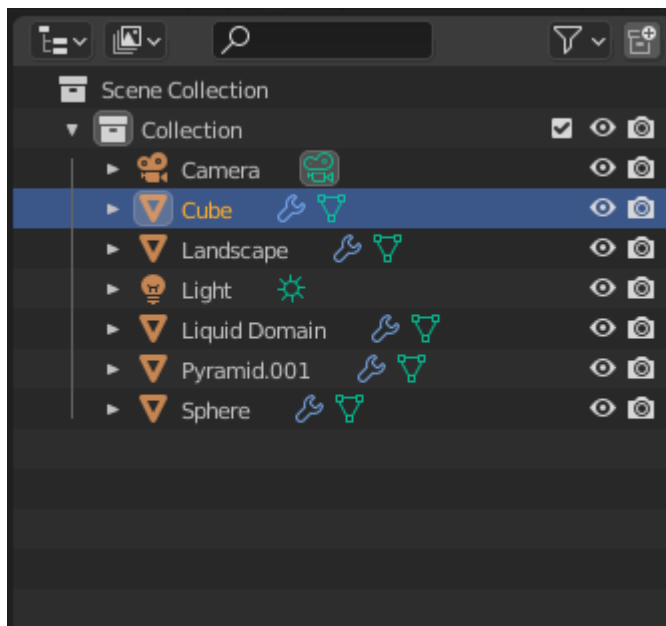
Desno od prikaza orijentacije nalazi se dosta koristan izbornik pomoću kojega možemo precizno pozicionirati odabrani objekt ili model u prostoru u odnosu na njegov odabrani centar ravnoteže ili centralnu točku koja se također može izmijeniti u izborniku sa Slike 2, *Object > Set Origin. Location*, pozicioniranje objekta po X,Y,Z osima. *Rotation*, rotacija po X,Y,Z osima. *Scale*, omogućuje razvlačenje ili skupljanje objekta po X,Y,Z osima. *Dimensions*, omogućuje preciznu promjenu dimenzija objekta po X,Y,Z osima.



*Slika 3 dio sučelja objašnjenog u tekstu iznad slike*

Desno od prethodno opisanog dijela sučelja nalazi se prostor koji sadrži navedene sve objekte koje imamo u sceni. Objekte možemo razvrstati u grupe što je korisno kog većih scena gdje nam omogućuje lakše snalaženje. Desno od naziva objekta nalaze se ikone sa

okom i kamerom pomoću kojih može maknuti objekt iz 3D prikaza ako ugasimo ikonu oka ili iz rendera ako ugasimo ikonu kamere.

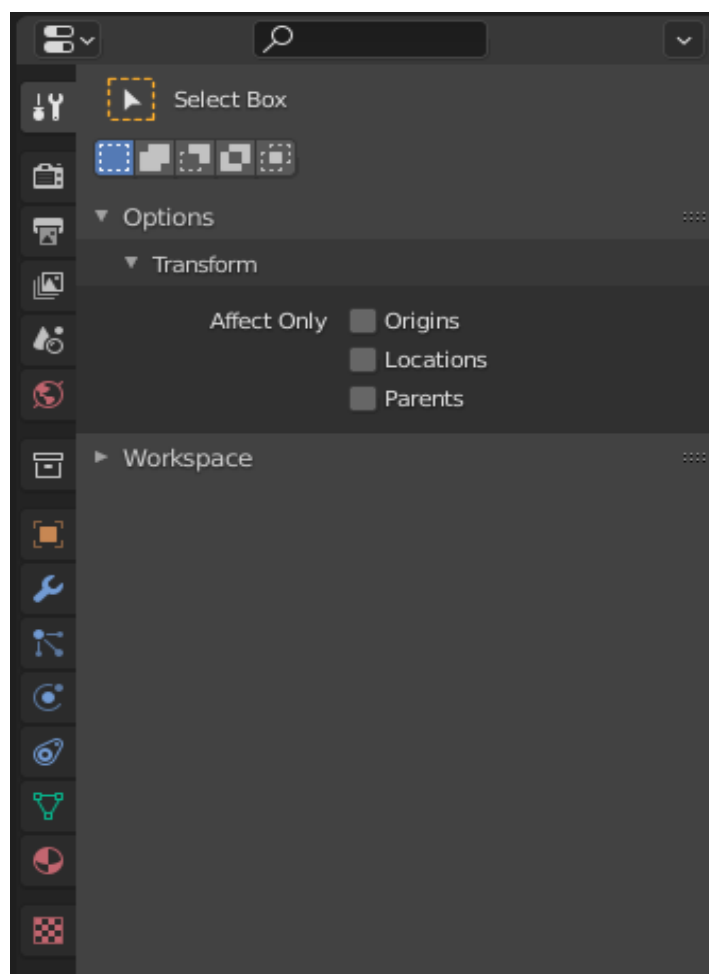


Slika 4 Dio sučelja objašnjen u tekstu iznad slike

Ispod prethodno opisanog dijela sučelja nalazi se izbornik *Properties*. Ovdje se podešavaju razne stvari od postavki same scene, rendera, objekta te ostalog kroz 15 izbornika koji su redom, od otvorenog izbornika na Slici 5 nadalje:

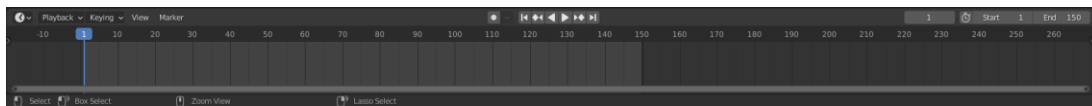
- Active tool and workspace settings – postavke selekcije objekata u 3D prostoru
- Render properties – postavke rendera ili finalnog proizvoda, odabir engine-a za render, komponente koja će provesti render (CPU ili GPU) i ostalo
- Output properties – postavke same datoteke rendera, rezulucija slike ili videa, postavke boje, odabir formata rendera i ostalo
- View layer
- Scene properties – postavke scene, odabir mjernih jedinica, podešavanje gravitacije i ostalo
- World properties – postavke pozadine, moguće promijeniti boju pozadine, napraviti prikaz neba ili kao pozdainu postaviti sliku od 360° te ostale postavke

- Collection properties
- Object properties – postavke objekta ili modela
- Modifier properties – postavke *modifier*-a
- Particle properties – postavke particle-a ili čestica
- Physics properties – postavke i odabir fizikalnih osobina koje možemo dodati objektima
- Object constraint properties
- Object data properties
- Material properties
- Texture properties



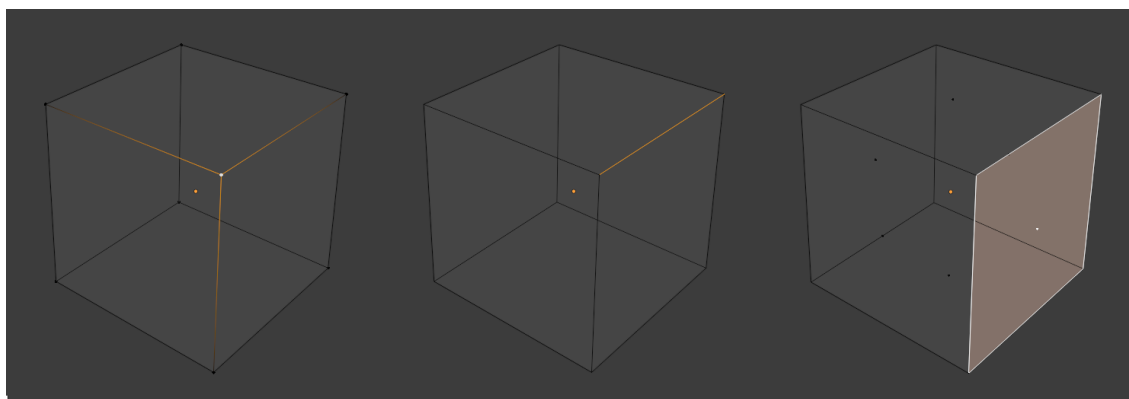
*Slika 5 Dio sučelja objašnjenog u tekstu iznad slike*

Na dnu sučelja ispod 3D prikaza nalazi se *Timeline* ili vremenska traka. Vremenska traka je radni okvir bez kojega ne možemo napraviti simulaciju. U njoj određujemo koliko *frame*-ova želimo renderati. Finalni render obično ima 24 fps, tako da je za video od 30 sekundi potrebno renderati 720 *frame*-ova. U njoj također testiramo samu simulaciju u fazi izrade, *bake*-amo željene postavke moramo stisnuti *play* na vremenskoj crti kako bi se isprobale promijenjene postavke.



*Slika 6 Vremenska crta*

Poznavanje ovih dijelova sučelja je dovoljno za izradu ovakvih jednostavnijih scena. Blender sadrži mnoštvo drugih alata, izbornika i workspace-ova koji nisu spomenuti te koji se koriste za druge potrebe ili onih koji zahtijevaju napredno poznavanje Blender-a. Dio koji je potrebno nadodati prethodnom je struktura trodimenzionalne geometrije s kojom se radi. Osnova svega je *vertex*, točka. Kada se dvije točke povežu dobije se *edge*, crta ili brid. Zadnji element je *face*, minimalni uvjet za dobivanje face-a je kada 3 točke zatvaraju trokut, a uobičajan face se sastoji od 4 točke koje zatvaraju kvadrat, pravokutnik ili romb. Neki programi koji koriste 3D modele zahtijevaju da model bude načinjen isključivo od trokuta ili *tris*-a kao oblika *face*-a. Pristup *vertex*-u, *edge*-u i *face*-u je moguć jedino u *Edit mode*-u i to samo za objekt koji smo prije prelaska u *Edit mode* selektirali u *Object mode*-u.



*Slika 7 Lijevo je selektiran Vertex, u sredini Edge te desno Face*

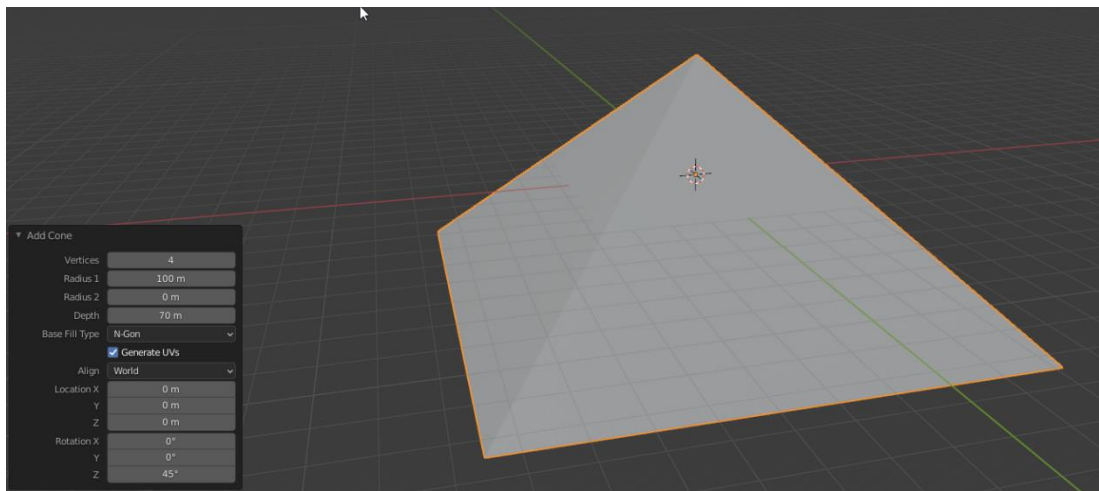
## 3. IZRADA OKRUŽENJA I SCENE

### 3.1 Izrada piramide

Scena se sastoji od piramide sa 4 otvora pri vrhu, pijeskom kao podlogom, kamenjem okolo piramide, kanalom te okruglim kamenom u kojemu će biti vatra.

Sve počinje sa otvaranjem novog dokumenta i odabirom standardnog početnog sučelja i okruženja što postavlja sivu kocku u nultu točku 3D sustava. Kocka se briše i dodaje se stožac pomoći izbornika *Add >Mesh >Cone*. Kako bi dobili piramidu kod dodavanja *Cone*-a mijenjamo postavke iz *popup* izbornika.

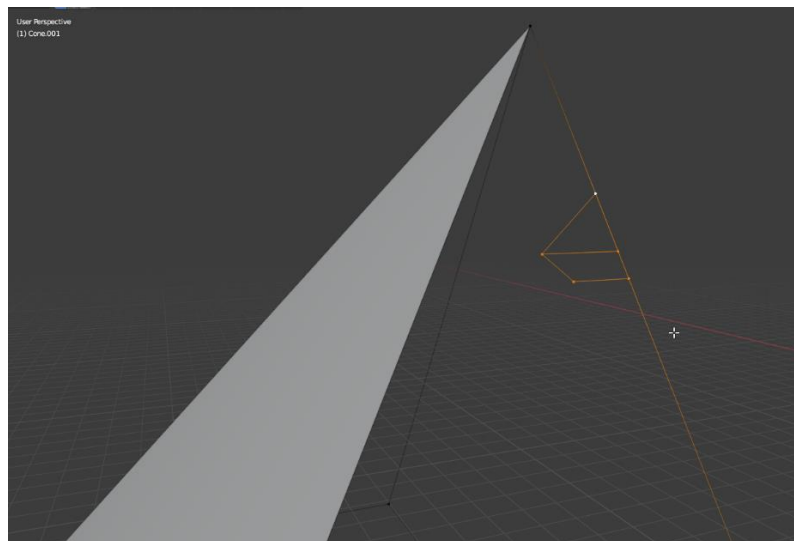
Vertices označava broj krajnjih točaka stožca te se postavlja na 4 kako bi dobili piramidu, Radius 1 je dužina stranice 100 m, Depth je visina 70 m. Rotation po Z osi postavljamo na 45°



Slika 8 Izrada piramide pomoću komande za dodavanje stožca, *Add >Cone*

Piramidu zatim dijelimo na pola tako da donji brid podijelimo na pola selektiranjem istoga zatim stisnemo desni klik na mišu te se otvara *popup* izbornik *Edge Context menu* u kojemu odaberemo prvu opciju na listi, a to je *Subdivide* koja dijeli brid na dva jednaka dijela sa *vertex*-om ili točkom u sredini. Nakon toga spajamo taj *vertex* na sredini brida sa vrhom piramide kako bi se dobio *edge* ili brid po visini piramide te se opet primjenjuje *Subdivide* na taj brid. Dobiveni *vertex* se sada nalazi na sredini visine piramide, a treba se nalaziti pri vrhu. Potrebno je pomaknuti *vertex* po bridu i istom kutu piramide, kako bi se to ostvarilo selektiramo *vertex* te stisnemo dva puta G na tipkovnici, pritiskom na G jednom omogućuje pomicanje točke bilo gdje, a pritiskom na G dva puta pomicanje se zaključava na os na kojoj se trenutno nalazi. *Vertex* se pomiče

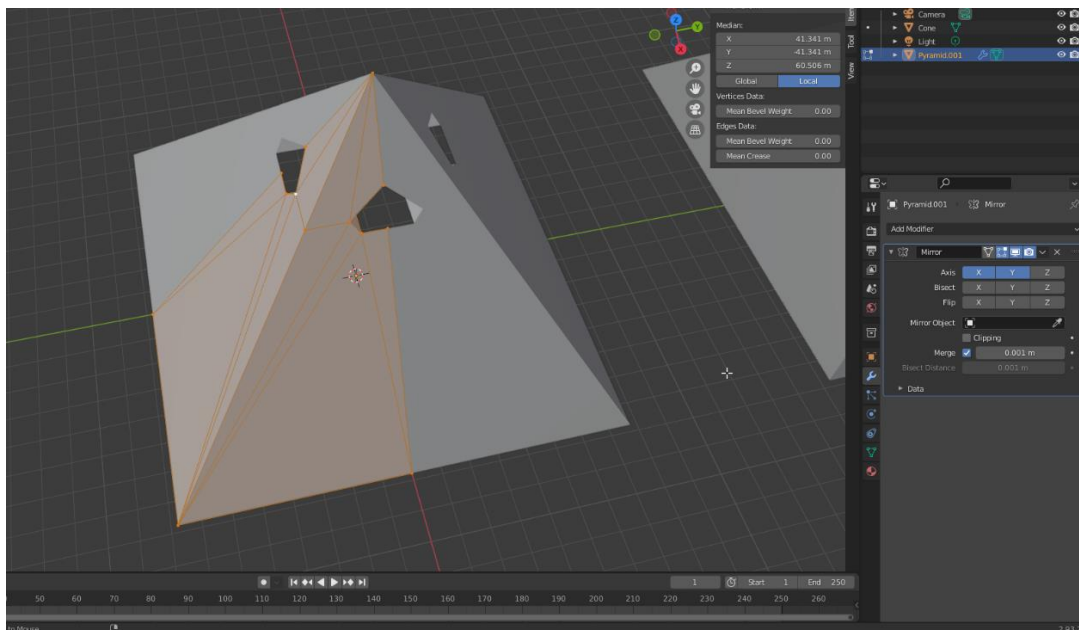
prema vrhu piramide, nakon pozicioniranja na željeno mjesto pritisnemo E pa zatim Y na tipkovnici. Sa E pozivamo komandu *extrude*, a Y zaključava pomicanje samo na Y os. *Extrude* izvlači iz selektiranog *vertex*-a još jedan i spaja ih sa *edge*-om. Na ovaj način će se napraviti otvor u piramidi. Za ostale rubne točke otvora ponavljamo proces selektiranja *edge*-a, primjenu komande *subdivide* te izvlačenje *vertex*-a po y osi u željeni oblik otvora.



Slika 9 Otvor u piramidi sa označenim *vertex*-ima

Otvor je na mjestu gdje bi trebao biti te je potrebno obrisati ostatak piramide i zatvoriti polovicu stranice piramide u kojoj je otvor sa *face*-ovima. To radimo tako da selektiramo dva *vertex*-a sa rubova piramide i jedan sa otvora te pritisnemo F na tipkovnici što je komanda za dobivanje *edge*-a ili *face*-a. Tri selektirane točke daju *face* te tako zatvaramo prazni prostor i dobivamo vanjsku površinu piramide.

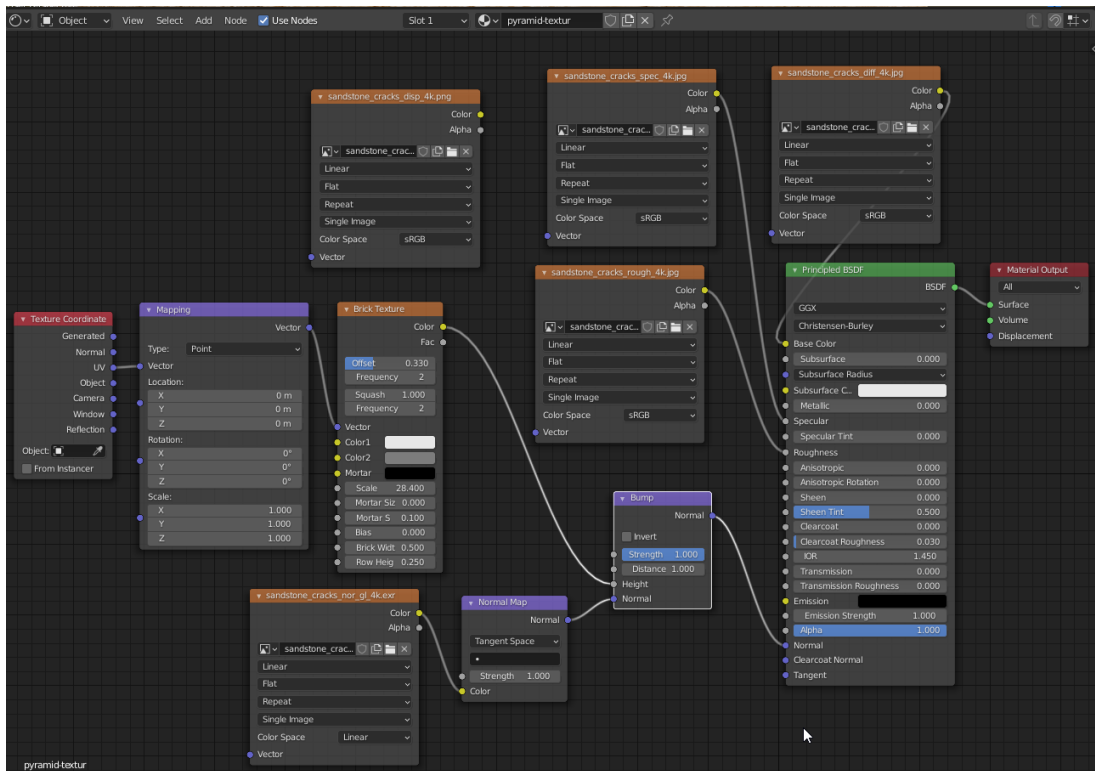
Kako bi se ponovno dobila piramida iz polovice jedne stranice piramide koristiti će se *modifier Mirror*. Njega nalazimo u izborniku *Properties* u krajnjem donjem-desnom dijelu sučelja. *Mirror modifier* zrcalno preslikava objekt po X, Y ili Z osi. Prvo preslikavamo po X osi kako bi se dobila četvrtina piramide. Sve *modifier*-e je moguće koristiti jedino u *Object mode*-u iako su vidljivi u *Edit mode*-u. Kako bi potvrdili *modifier* moramo biti u *Object mode*-u te kliknuti na logo sa kvačicom i odabrati *Apply*. Nakon dobivanja četvrtine piramide opet se koristi *modifier Mirror*, ovoga puta po X i Y osi te se dobiva cijela piramida.



Slika 10. Prikaz piramide prije potvrđivanja Mirror modifier-a

Kako bi piramida dobila nešto realniji izgled od sive boje moramo dodati teksturu. Prvo selektiramo piramidu u *Object mode*-u, zatim prelazimo u *Shading workspace* koji se nalazi u gornjem dijelu sučelja. Sučelje je ovdje podijeljeno na dva dijela, gore je 3D prikaz, a dolje *Shader editor* gdje radimo sa *node*-ovima. *Node*-ovi se mogu objasniti kao vizualni prikaz matematičkih operacija, u principu preko njih se kroz razne parametre mogu dobiti realni prikazi tekstura drveta, kamena, vode, betona i svih ostalih materijala ili površina.

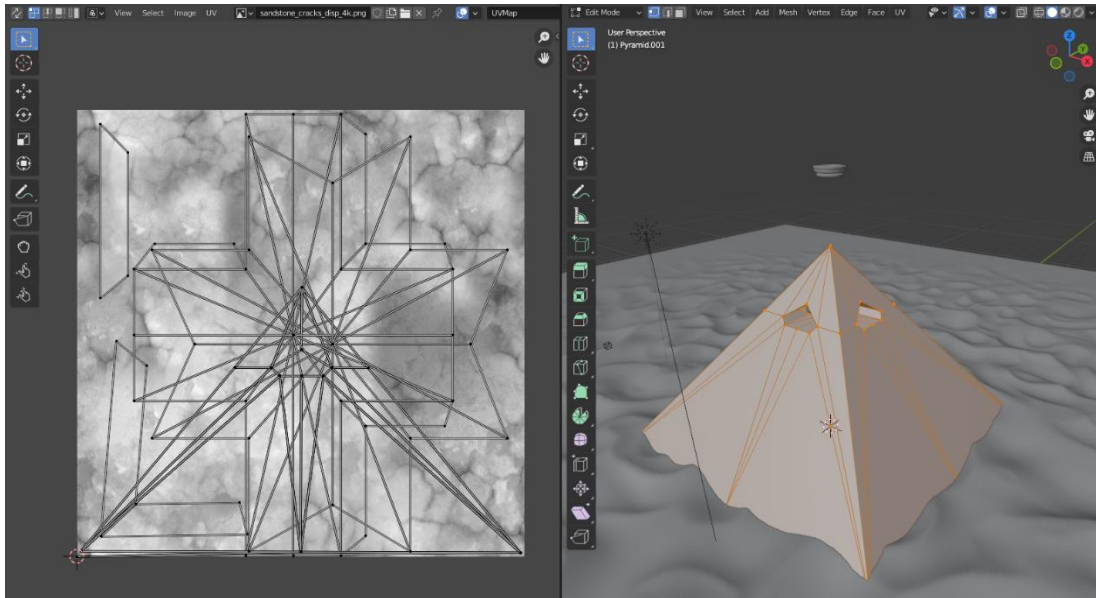
Prilikom dodavanja novog materijala odmah dobivamo *Principled BSDF* i *Material output* *node*-ove te *output BSDF* spajamo na *Surface input* kod *Material output* *node*-a. Zatim dodajemo prethodno preuzete teksture sa interneta u *Shader editor* i spajamo ih na *Principled BSDF* *node* u odgovarajuća mjesta. Za spajanje normal mape potrebno je dodati *Bump* i *Normal Map* *node*-ove. *Normal map* teksturu preko *Color output*-a spajamo na *Color input* na *Normal Map* *node*-u. *Normal Map* *output* *Normal* spajamo na *Normal input* na *Bump* *node*-u te *output* sa *Bump* *node*-a spajamo na *Normal input* na *Principled BSDF* *node*-u. Za dobivanje teksture cigli po površini piramide dodaje se *Brick texture* *node* te spajamo njegov *Color output* u *Bump* *node* *Height input*. Za mijenjanje veličine cigli u teksturi dodaje se *Node Wrangler* *plugin* na *Brick texture* *node* te sa *Texture Coordinate* *node*-a *UV output* spajamo na *Vector input* *Mapping* *node*-a. Sada se vrši podešavanje nekoliko parametara i tekstura piramide je gotova.



*Slika 11 Prikaz Shader editor-a sa svim node-ovima koji se koriste za teksturu piramide*

Kako bi teksturu primijenili na piramidi potrebno je napraviti *UV unwrap* to jest površine objekta prenijeti u 2D oblik kako bi se tekstura sa slike prenijela na objekt. Potrebno je otići u *UV Editing workspace* koji se nalazi u gornjem dijelu sučelja. U *Edit mode*-u koji nas dočeka sa desne strane označimo cijeli objekt, pritisnemo U na tipkovnici i odaberemo *Unwrap* iz izbornika. S lijeve strane učitamo sliku teksture koju koristimo i možemo vidjeti 3D objekt razdijeljen u dijelove u 2D obliku. Moguće je poredati dijelove modela jedne na druge kako bi se što bolje iskoristila rezolucija same teksture. Pošto su sve četiri stranice piramide jednake one su jednostavno posložene jedna na drugu, dijelovi unutrašnjosti piramidu u otvoru preko njih.

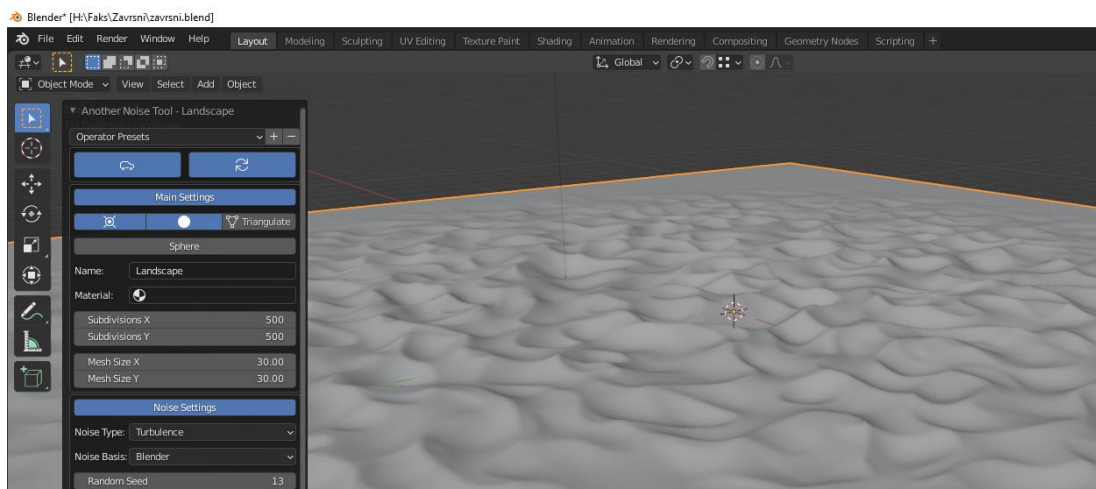




Slika 12 UV unwrap modela piramide

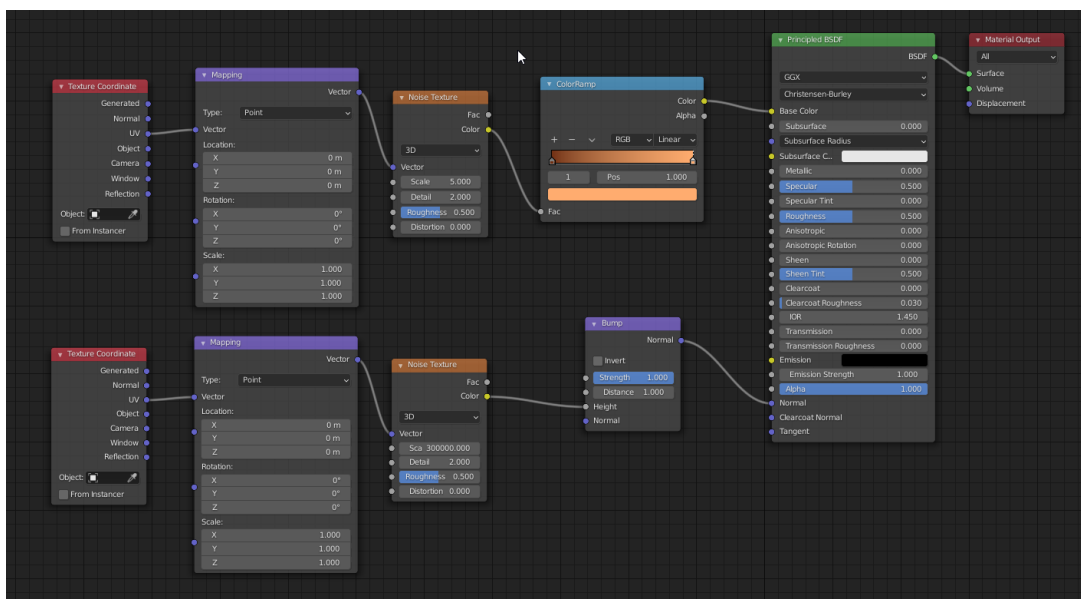
### 3.2 Izrada pješčane podloge

Pustinjski pijesak je najprirodnije okruženje piramidi te se najbolje uklapa u trenutnu scenu. Izrada pustinjskog pijeska započinje dolaskom u *Edit* izbornik u gornjem lijevom kutu sučelja gdje odabiremo stavku *Preferences*. Zatim se odabire izbornik *Add-ons* gdje uključimo: *Add Mesh: A.N.T.Landscape*. U scenu se zatim dodaje novi objekt: *Add >Mesh > Landscape*. U izborniku koji se otvorio potrebno je kliknuti na *Operator present* te odabrati *Dunes* i promijeniti: *Subdivisions X* na 500, *Subdivisions Y* na 500, *Mesh size X* na 30 i *Mesh size Y* na 30.



Slika 13 Prikaz pješčane podloge bez teksture

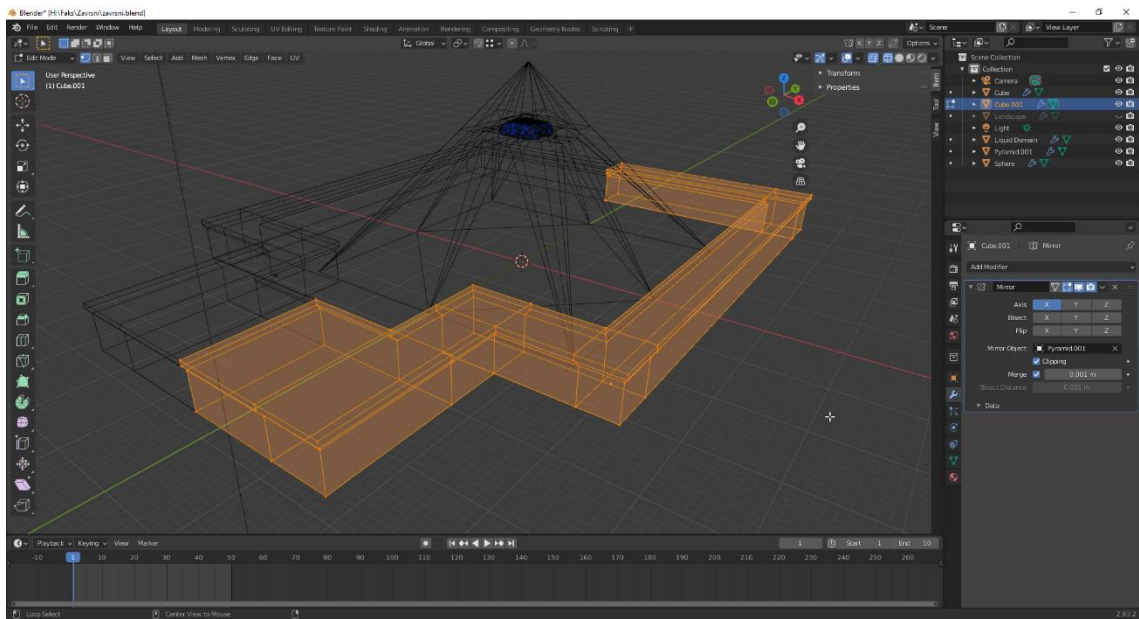
Teksturiranje pješčane podloge se kao i kod piramide radi u *Shader workspace*-u. Nakon prelaska u *workspace* dodajemo novi materijal. Teksturu pijeska ćemo dobiti isključivo preko *node*-ova bez slike kao podloge teksture. Za početak spajamo *BSDF output* na *Surface input Material output node*-a. Dodajemo *Noise texture* i *ColorRamp* *node*-ove. *Color output* od *Noise texture node*-a spajamo na *Fac input ColorRamp node*-a. *Color output* od *ColorRamp node*-a spajamo na *Base Color input Principled BSDF node*-a. Zatim se na kratko prebacujemo u *UV Editing workspace*, označimo sav cijeli model pijeska, pritisnemo U na tipkovnici te odaberemo: *Project from view*. Vraćamo se u *Shading workspace*, u *ColorRamp node*-u mijenjamo crnu i bijelu boju u tamno smeđu i blijedu žutu to jest prirodnu boju pijeska. Na *Noise Texture node* dodajemo *Node Wrangler* te kod *Texture Coordinate node*-a prebacujemo *output* sa *Generated* na *UV*. Kako bi se dobila tekstura pijeska u obliku znaca kopiramo *Noise Texture*, *Mapping* i *Texture Coordinate node*-ove i dodajemo *Bump node*. *Noise Texture Color output* spajamo na *Height input Bump node*-a, a *Normal output* iz *Bump node*-a spajamo na *Normal input Principled BSDF node*-a. Na kraju je samo potrebno izmijeniti *Scale* parametar kod *Noise Texture*-a koji predstavlja normal mapu na veliki broj od 300000 čime dobivamo zrnatu teksturu pijeska.



Slika 14 Prikaz Shader editor-a sa svim node-ovima koji se koriste za teksturu pijeska

### 3.3 Izrada zidina

Zidine će okružiti piramidu i služiti kao granica do koje voda dolazi. U scenu dodajemo kocku: *Add > Mesh > Cube*. Postavljamo ju na mjesto gdje će biti rub zida te odmah uključujemo *Mirror modifier* po X osi. Pomoću komandi *Extrude* pritiskom na tipku E i *Scale* pritiskom na tipku S na tipkovnici izvlačimo i podebljavamo gornji rub zida. Kada smo dobili osnovni oblik zida opet koristimo komadu *Extrude* i povlačimo zid oko piramide. Kako bi se dobila tekstura zida koristi se isti materijal kao i kod piramide, potrebno je samo provesti *UV unwrap*.



Slika 15. Prikaz zidina prije potvrđivanja *Mirror modifier*-a

### 3.4 Izrada postolja za vatru i objekta s kojim se dim sudara

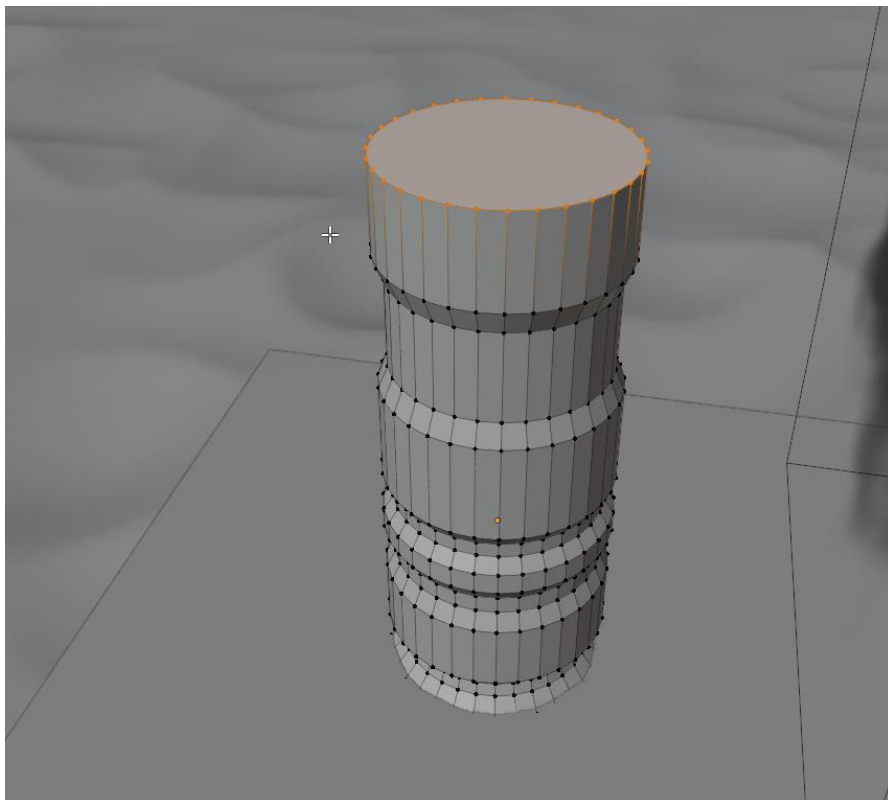
Postolje ćemo napraviti tako da dodamo kuglu *Add > Mesh > UV Sphere* te zatim sa komandama *Scale*, S na tipkovnici i *Extrude*, E na tipkovnici formiramo željeni oblik kugle. Polovicu kugle smo uklonili jer je višak. Za teksturu koristimo isti materijal kao i za piramidu samo bez teksture cigli.

Stupove radimo na način da u scenu dodamo valjak *Add > Mesh > Cylinder*. Valjak zatim oblikujemo pomoću komandi *Scale*, S na tipkovnici i *Extrude*, E na tipkovnici dok ne dobijemo izgled obrađenog stupa. Materijal je isti kao i za postolje.

Objekt s kojim se dim sudara dobijemo tako da u scenu dodamo stožac sa četiri strane, to jest piramidu *Add > Mesh > Cone* te upišemo 4 strane. Objekt zatim oblikujemo sa komandama *Scale*, S na tipkovnici i *Extrude*, E na tipkovnici. Materijal ostaje isti.



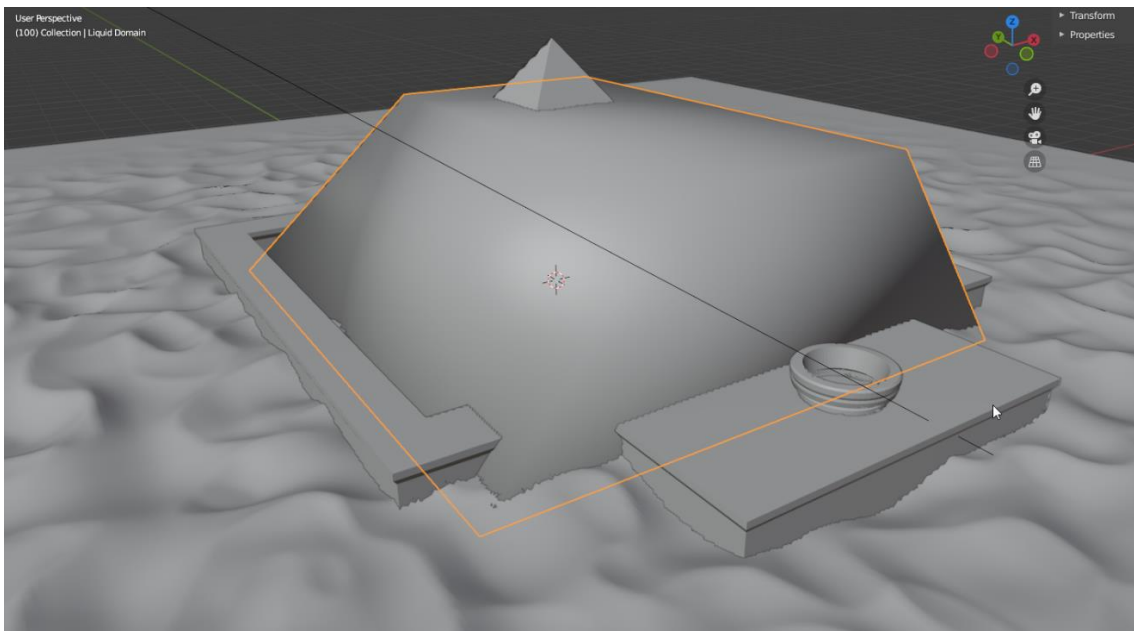
*Slika 16. Postolje za vatru sa stupovima i objektom u koji se dim sudara*



*Slika 17. Prikaz stupa tijekom oblikovanja*

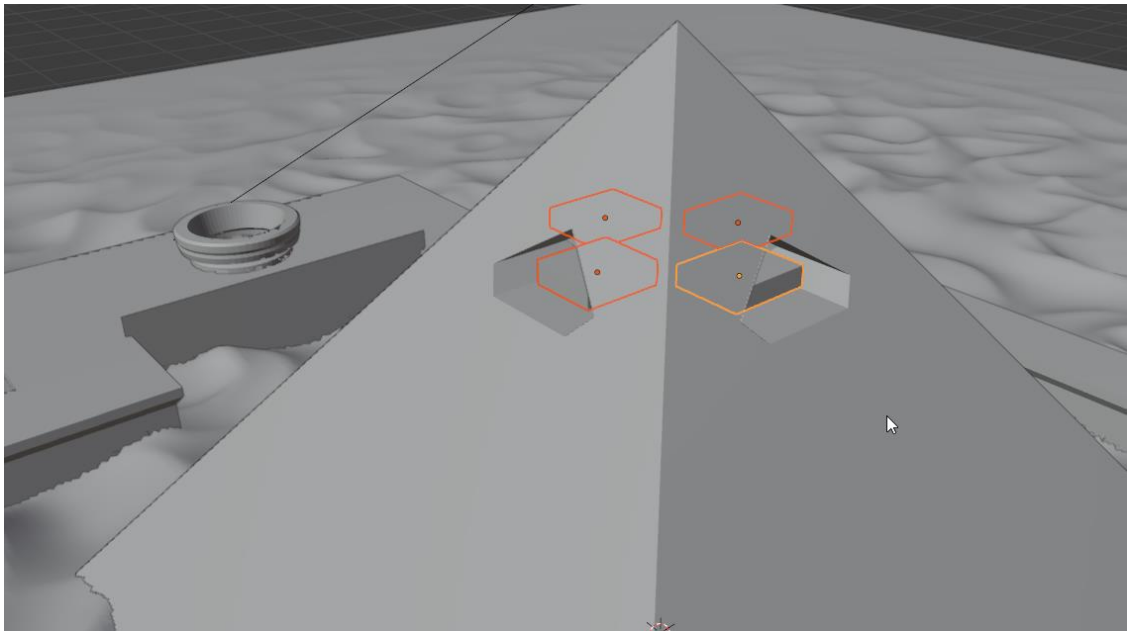
## 4. IZRADA SIMULACIJE FLUIDA

Simulaciju fluida započinjemo dodavanjem kocke koja će postati domena simulacija, fluid se nalazi samo u toj domeni. Dodavanje kocke: *Add > Mesh > Cube*. Kocku je potrebno podesiti točno u prostor u kojemu želimo kretanje fluida, bitno je da je taj prostor što je moguće manji. Kada je kocka na mjestu odlazimo u izbornik *Properties* te na izbornik *Physical Properties* gdje odabiremo *Fluid* kao fizikalno svojstvo koje želimo dodati objektu. Zatim pod *Type* odabiremo *Domain* i pod *Domain type* odabiremo *Liquid*.



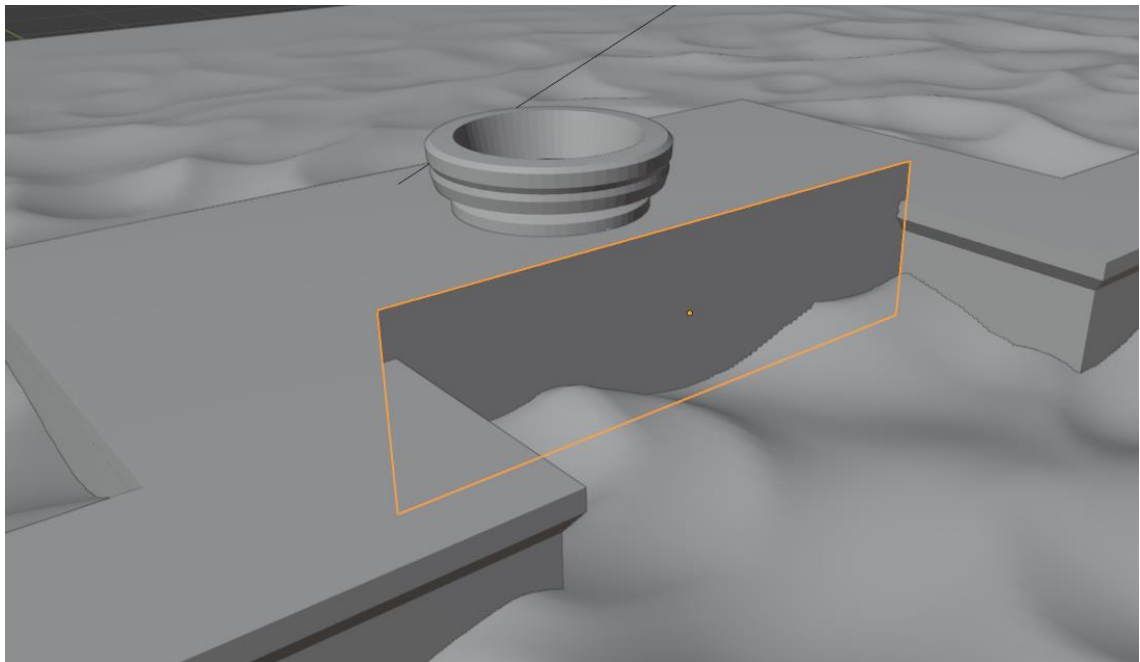
Slika 18. Pozicioniranje Domain-a ili domene fluida u scenu

Kada imamo domenu dodajemo u njenim granicama još jednu kocku koja će biti izvor vode ili *Inflow*. Pretvaranje kocke u izvor je slično kao i kod domene pa pod *Type* odabiremo *Flow*, *Flow Type* odabiremo *Liquid* i kod *Flow Behavior* odabiremo *Inflow*. Za ovu scenu će se koristiti 4 ista *Inflow* objekta što se kroz preko sto pokušaja i pogrešaka pokazalo kao najbolja varijanta izvora. Izvori se nalaze u otvorima pri vrhu piramide i iz njih će fluid teći prema dnu gdje će ispuniti prostor između zidina.



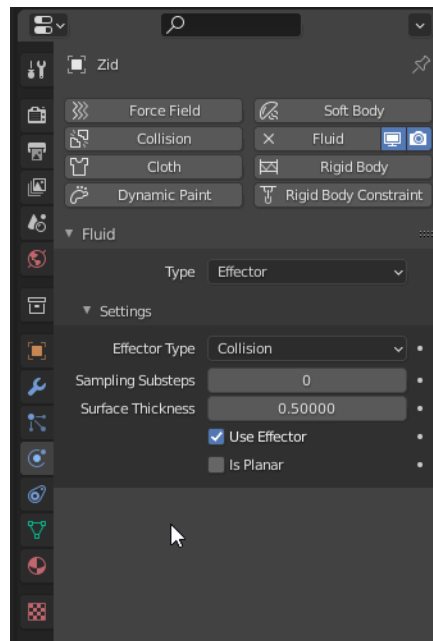
Slika 19. Pozicioniranje Inflow-a ili izvora fluida u scenu

U scenu ćemo dodati i odvod vode ili *Outflow*. Odvod će umjesto kocke biti ravna ploha, dodajemo ju preko *Add > Mesh > Plane*. Promjenu u odvod vršimo isto kao domenu i izvor pa pod *Type* odabiremo *Flow*, *Flow Type* odabiremo *Liquid* i kod *Flow Behavior* odabiremo *Outflow*. Odvod je pozicioniran na kraju zidina u uvučenom dijelu te do njega voda zadnja dolazi i višak nestaje iz scene.



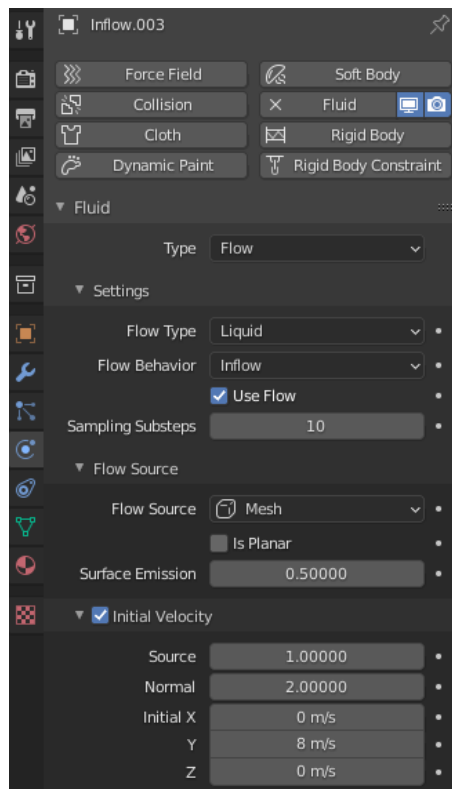
Slika 20. Pozicioniranje Ourflow-a ili odvoda fluida u sceni

Iduća stavka koju treba napraviti je odrediti objekte koji su u dodiru sa fluidima što su u ovoj sceni skoro svi. To radimo isto kao i za ostale objekte fluida pa idemo u izbornik *Properties* te na izbornik *Physical Properties* gdje odabiremo *Fluid* kao fizikalno svojstvo koje želimo dodati objektu. Zatim pod *Type* odabiremo *Effector* i pod *Effector Type* odabiremo *Collision*. Također je bitna stvar pod *Surface Thickness* upisati vrijednost od minimalno 0.5 jer se pokazalo da kod manjih vrijednosti fluid prolazi kroz objekt umjesto da se sudara sa njime.



Slika 21. Prikaz postavki za *Effector*-e ili objekte sa kojima se fluid sudara

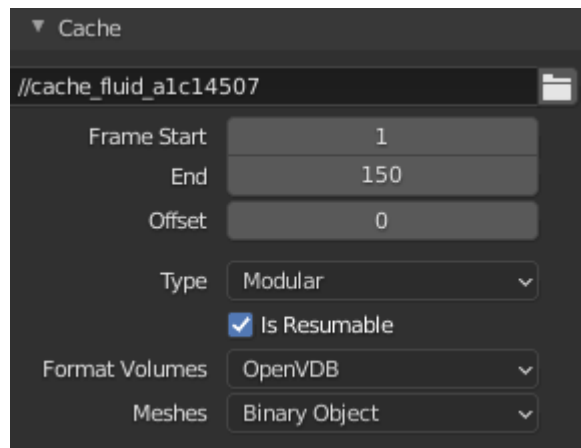
Svi objekti su na svojim mjestima i pod postavkama fluida te sada možemo napraviti detaljne postavke same simulacije. Počinjemo od *Inflow* objekata. U fizikalnim postavkama u kojima smo već podesili tip fluidnog objekta ispod *Flow Behavior*-a nalazi se kvačica pored koje piše *Use Flow*. Preko nje određujemo kada fluid iz izvora teče, a kad ne, što se koristi kada želimo da fluid počne teći poslije početka animacije ili kada želimo da usred animacije tok prestane. To radimo tako da prilikom *bake*-a simulacije zaustavimo *bake* i ugasimo *Use Flow* te nastavimo *bake*. Ispod *Use Flow*-a nalazi se *Sampling Substeps* koji postavljamo na vrijednost 10 što poboljšava kvalitetu brzih tokova fluida. *Surface Emission* postavljamo na 0.5 što objektu daje dodatnu nevidljivu debljinu koja je bitna kako fluid ne bi prošao kroz objekt. Uključujemo *Initial Velocity* te *Source* ostavljamo na 1.0, *Normal* postavljamo na 2.0 te za četiri *Inflow* objekta svakome drugačije ubrzanje *Initial X* ili *Initial Y* ubrzanja po osima na  $-8\text{m/s}^2$  ili  $8\text{m/s}^2$  ovisno na koju stranu piramide trebaju puštati fluid.



Slika 22. Prikaz postavki za Inflow objekte ili izvore fluida

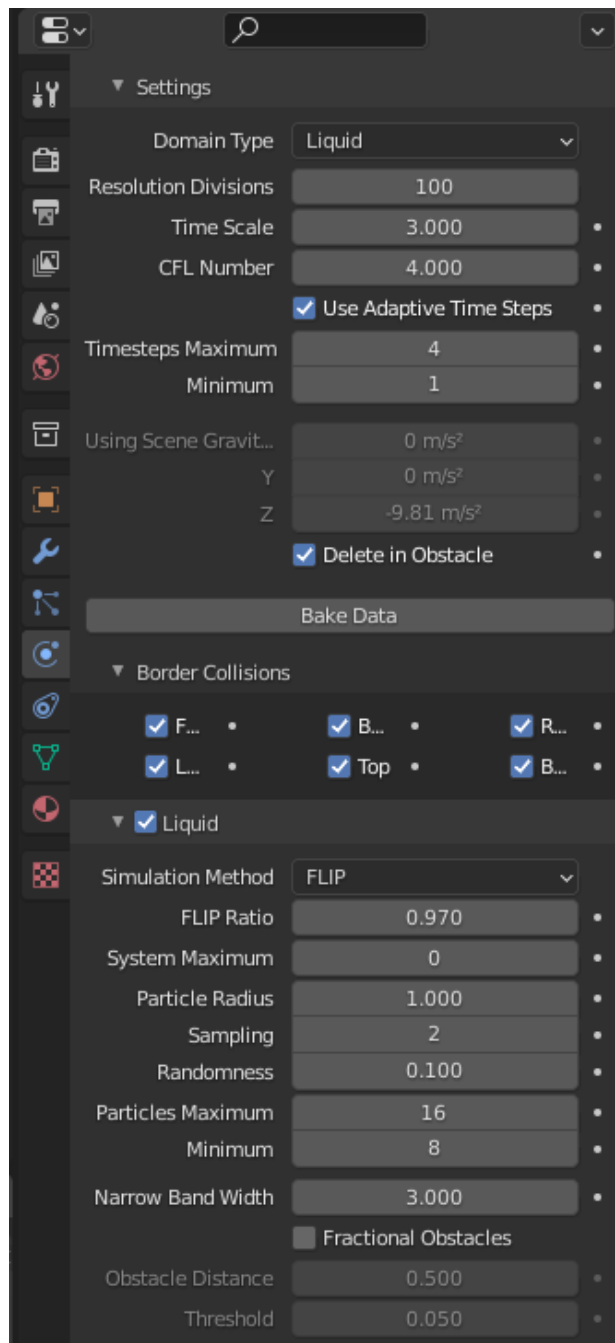
Domena simulacije se pretvara u fluid kada sve podesimo, *bake*-amo i pustimo animaciju. Kako bi dobili simulaciju približno onakvu kakvu smo zamislili potrebno je bezbroj puta mijenjati postavke domene i izvora te svaki puta *bake*-ati simulaciju kako bi smo uopće vidjeli što smo promijenili. Prije nego što počnemo sa postavkama potrebno je na dnu izbornika za domenu podesiti *Cache* postavke, *Cache* je privremena memorija koju koristi simulacija prilikom *bake*-a. *Frame start* označava prvi *frame* od kojega će simulacija početi dok *End* označava kraj, bitno je postaviti *End* u dovoljan broj *frame*-ova kako bi se simulacija provela u cijelosti. *Type* označava način pregleda simulacije, postoje dva, *Replay* i *Modular*. *Replay* radi na način da u stvarnom vremenu, kada stisnemo *play* animacije na *timeline*-u, *bake*-a simulaciju sa svim njenim dijelovima koje smo odabrali te upravo zbog toga nekad daje nepouzdan rezultate. *Modular* radi na način da svaki dio po redu, *Liquid*, *Particles* i *Mesh*, moramo *bake*-ati zasebno što je možda malo sporije, ali su rezultati pouzdaniji te je prije finalnog *render*-a nužno postaviti *Type* na *Modular* i *bake*-ati svaki dio zasebno. Potrebno je također uključiti *Is Resumable* opciju koja omogućuje da nastavimo *bake* ako smo ga zaustavili bez da se cijeli *bake* mora početi iznova kao što je bio slučaj prije *Mantaflow*-a.





Slika 23. Postavke Cache-a

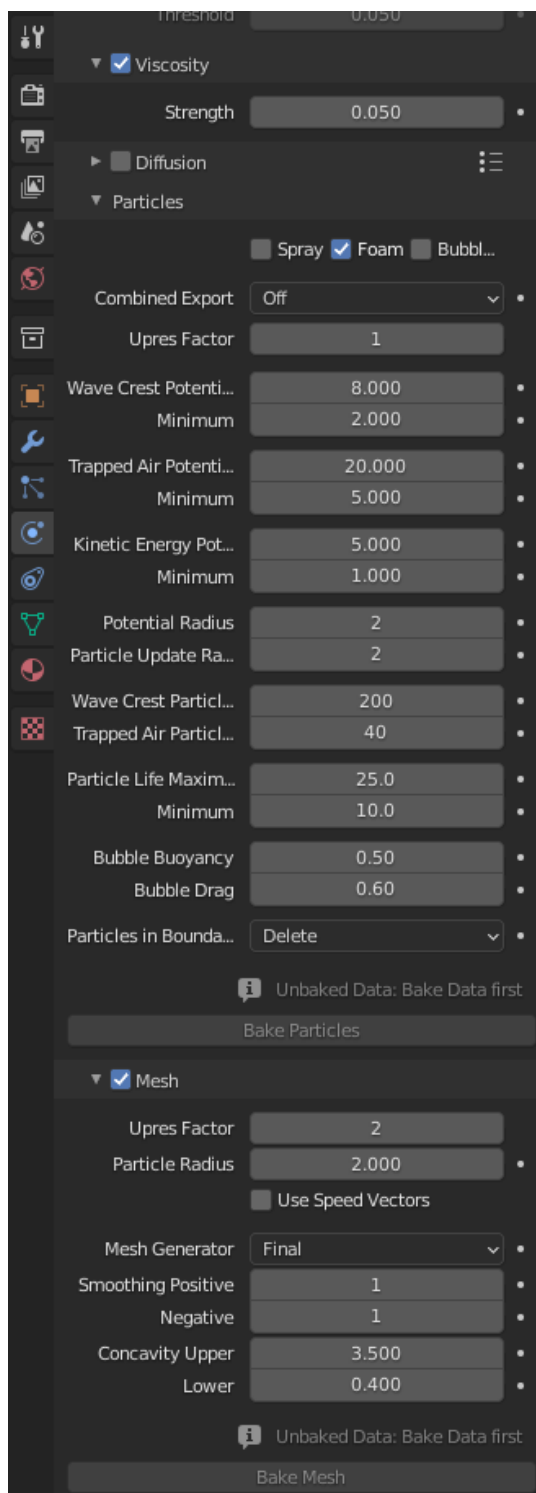
Zatim prelazimo na postavke domene koje najviše utječu na prikaz simulacije. Prva i najbitnija stavka je *Resolution Divisions* koju za finalni *render* postavljamo na 128, ali u principu ona bi trebala biti postavljena na 256 za realne simulacije fluida. *Resolution Divisions* označava podjelu najduže stranice domene na, u ovom slučaju, 128 dijelova. Dobiveni razmak se jednako odnosi na sve stranice neovisno o njihovoj dužini. To znači da se svaka domena pri *Resolution Divisions*-u od 128 dijeli na 128 dijelova po X osi, 128 dijelova po Y osi i 128 dijelova po Z osi. Tako se dobije, nama nevidljiva, 3D rešetka po kojoj fluidi formiraju svoje kretanje. Svaki put kada povećamo rezoluciju za 2 puta, na primjer sa 64 na 128, vrijeme *bake*-a se poveća za približno 8 puta. Rad na malim rezolucijama u većini slučajeva nije moguć zato što je često izvor bude manji od jedne ćelije domene te simulacija ne radi. Nakon rezolucije dolazi *Time Scale* gdje podešavamo brzinu simulacije, ovdje ju postavljamo na 3.0. Zatim imamo *CFL Number* koji postavljamo na 4.0, on označava maksimalnu brzinu po ćeliji. Ukljči se *Use Adaptive Time Steps*, *Timesteps maximum* se postavlja na 4 dok je *Minimum* na 1 što uzrokuje brži i lakši protok sa manje interakcije sa površinama objekta. Ispod imamo *Bake Data* opciju koju kliknemo kada želimo *bake*-ati postavke samo za simulaciju fluida. Postavke iz *Liquid* izbornika ostavljamo na zdanima. Tip simulacije fluida koji se koristi je *Flip Fluids*.



Slika 24. Domain i Liquid postavke

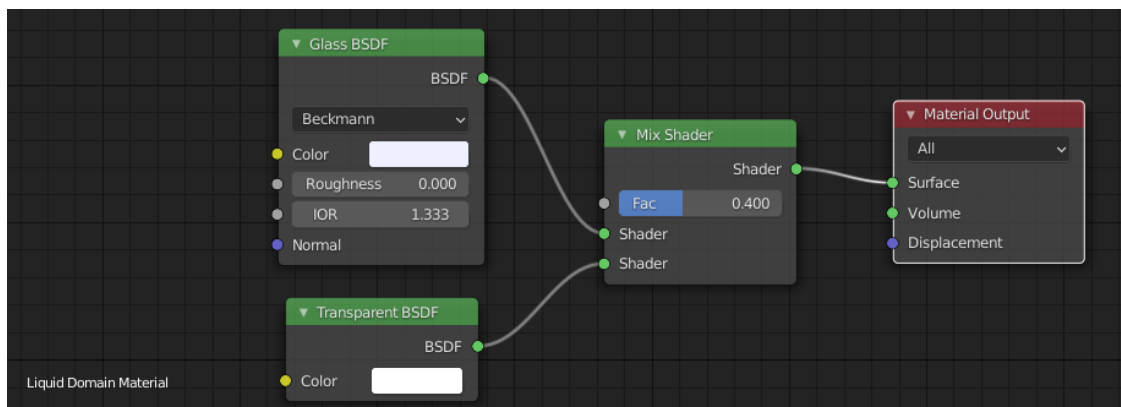
Kod simulacija vode dobro je upaliti *Viscosity* opciju zbog realnijeg kretanja fluida vode kada podesimo unutarnji koeficijent trenja tekućina. *Particles* omogućuje dodavanje čestica pjene, špricanja i mjehurića. *Spray* i *Bubbles* nisu toliko vidljivi pri render-u dok se *Foam* ipak više ističe te se zato koristi u sceni. *Particles* iziskuju velike resurse računala pošto su to 3D objekti koje sami dodajemo te je njih pri render-u mnogo i jako su male veličine. Njih zasebno *bake*-amo preko opcije *Bake particle* na dnu njihovog izbornika. *Mesh* postavke označavaju 3D model fluida koji se kao i svaki

drugi objekt u *Blenderi*-u sastoji od *vertex*-a, *edge*-eva te *face*-a. Najbitnija postavka kod *Mesh*-a je *Upres Factor* koji za uneseni iznos umnožava rezoluciju 3D modela fluida. Rezolucija koja se umnožava je *Reslution Divisions* sa početka postavki domene. *Upres Factor* ćemo postaviti na 2 za ovu simulaciju. Ispod postavki za *Mesh* imamo opciju *Bake Mesh* koji je također potrebno zasebno *bake*-ati.

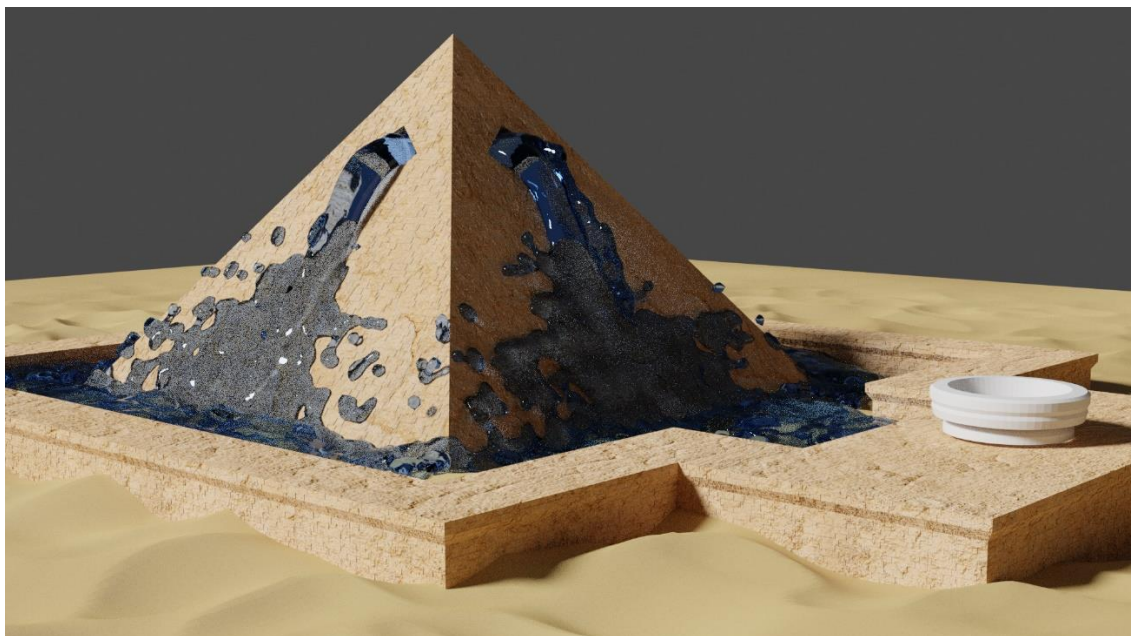


Slika 25. Postavke za *Viscosity*, *Particles* i *Mesh*

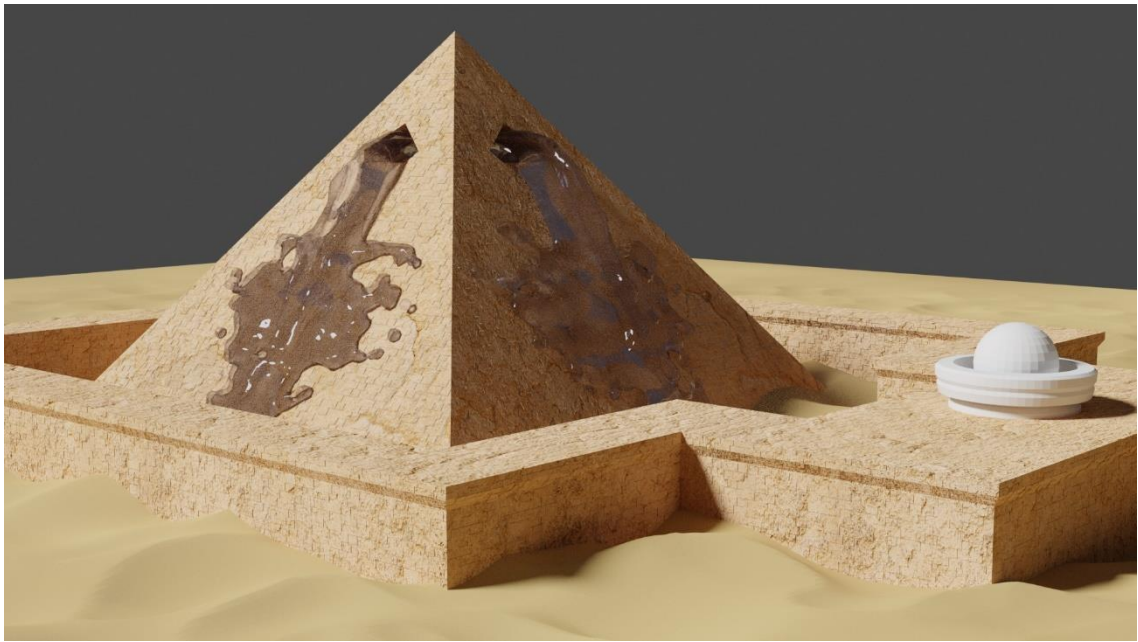
Domena postaje model fluida nakon *bake*-a te joj moramo dodati teksturu kako ne bi bila siva nego blizu boje vode to jest transparentna. Kako bi to ostvarili moramo otići u *Shading Editor* te dodati *Glass BSDF*, *Transparent BSDF* i *Mix Shader node*-ove.



Slika 26. Postavke za teksturu fluida



Slika 27. Render fluida u plavoj boji

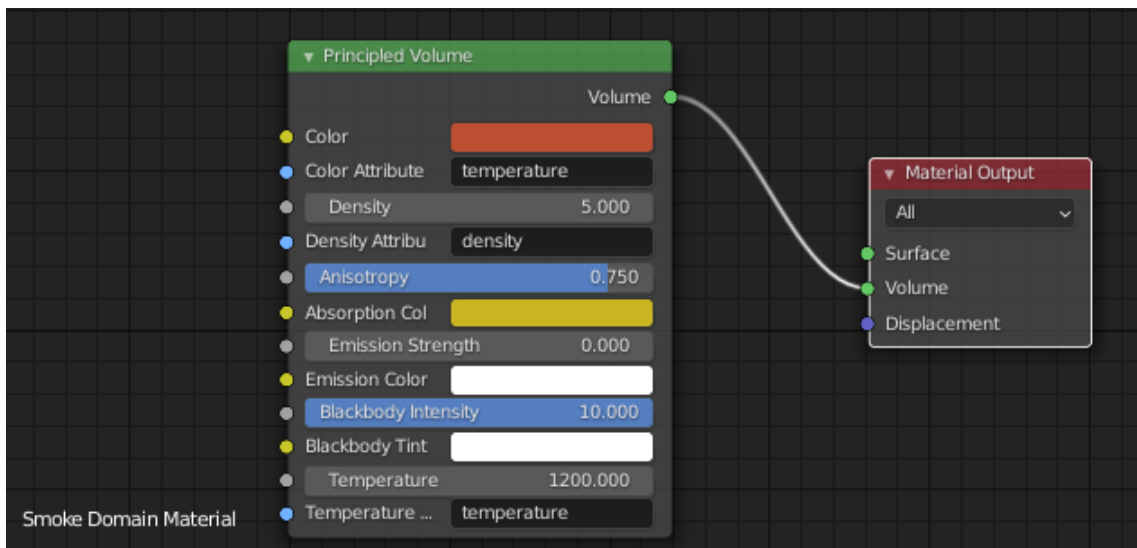


*Slika 28. Render fluida nalik na vodu, transparentna površina fluida*

## 5. IZRADA SIMULACIJE DIMA I VATRE

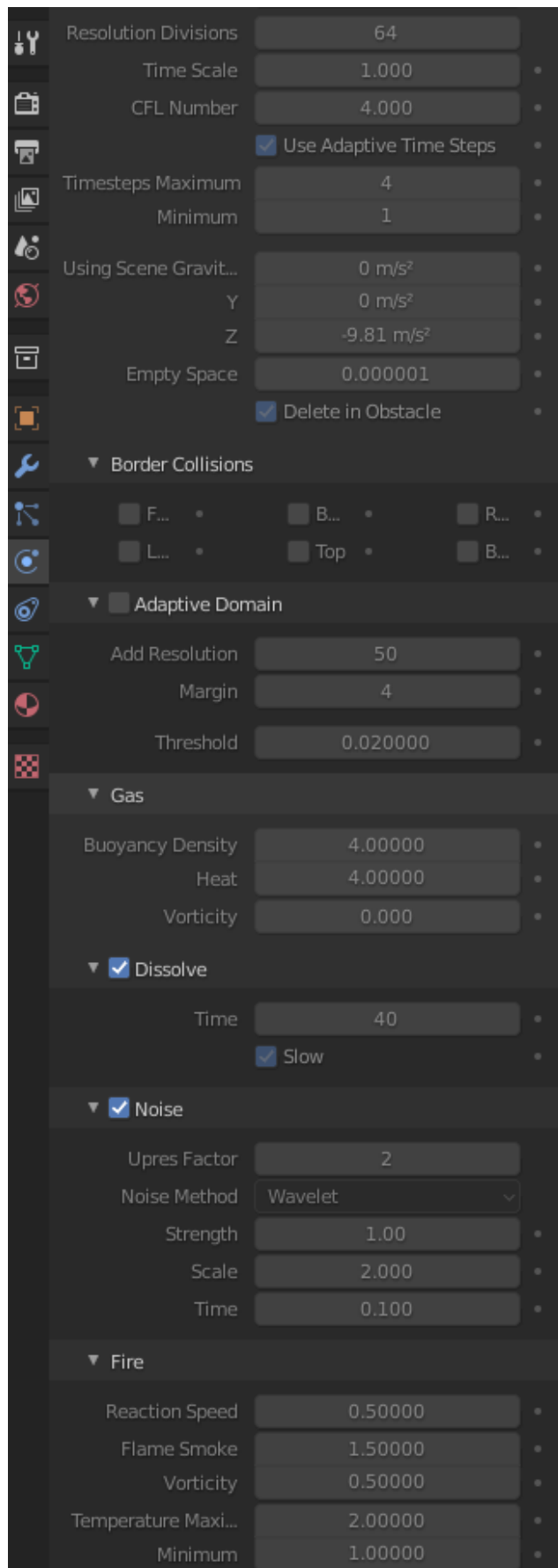
Simulaciju dima i vatre je uz *Mantaflow* fizikalni simulacijski okvir relativno jednostavno napraviti. Nije potrebno zasebno raditi simulaciju vatre pa dima jer postoji opcija *Fire and Smoke* koja poveže vatru i dim u cjelinu. Također je moguće zasebno napraviti i vatru i dim, ovisi što želimo. Simulacija dima se uz određene postavke može pretvoriti u maglu ili prikaz isparavanja tekućina, mogućnosti su mnogobrojne. Kako bi dodali vatru i dim moramo kao i kod simulacije fluida postaviti objekte koji će biti *Inflow* ili izvor, *Domain* ili domenu simulacije te *Outflow* ili odvod. *Inflow* će biti postavljen u kameno postolje ispred piramide koje se može vidjeti na Slici 26 iznad ovog teksta. Simulacija je zamišljena tako da dim dolazi do kamene prepreke iznad postolja i širi se okolo nje te nestaje.

Kao *Inflow* objekt koristi se kugla te ju dodajemo u scenu preko izbornika *Add > Mesh > UV Sphere*. Kuglu pozicioniramo u sredinu postolja i dodajemo objekt koji je biti domena simulacije a to je uvijek kocka, *Add > Mesh > Cube* te kocku pozicioniramo oko postolja i povećavamo sa komandom *Scale*, S na tipkovnici, kako bi dim imao dovoljno mjesta za širenje. Objektu sa kojim se dim sudara moramo dodati *Effector* postavku pa idemo u izbornik *Physical Properties*, odabiremo *Fluid*, pod *Type* odaberemo *Effector* te pod *Effector Type* odaberemo *Collision*. *Surface Thickness* postavimo na 0.5. Potrebno je također u *Shader editor*-u napraviti teksturu dima i vatre pa u *editor*-u dodajemo samo jedan *node*, *Principled Volume*. *Volume output* od *Principled Volume node*-a spajamo na *Volume output Material output node*-a. Kod *Principled Volume node*-a mijenjamo *Color* u naračasto-crvenu boju, *Color Attribute* upisujemo *temperature*, *Density* postavljamo na 5, *Anisotropy* postavljamo na 0.75, *Absorption Color* postavljamo u žutu, *Blackbody Intensity* postavljamo na 10 i *Temperature* na 1200.



Slika 29. Postavke teksture vatre i dima

Sada prelazimo na postavke te selektiramo *Inflow* i odlazimo u *Physical Properties* izbornik gdje od zadanih postavki samo mijenjamo *Fuel* na vrijednost 2. Selektiramo *Domain* te prvo postavljamo *Resolution Divisions* na 64, *Time Scale* na 1, *CFL Number* na 4, *Timesteps Maximum* na 4, *Minimum* na 1. Iz *Gas* izbornika *Buoyancy Density* i *Heat* postavljamo oboje na 4 kako bi se dim brže dizao u visinu. *Dissolve* postavljamo na 40 što označava vrijeme kada će se dim polako raspasti i nestati. Kod *Noise*-a *Upres Factor* postavljamo na 2 što povećava kvalitetu izgleda dima. Iz *Fire* izbornika *Flame Smoke* postavljamo na 1.5 što povećava količinu dima, *Temperature Maximum* postavljamo na 2 te *Minimum* na 1.



Slika 30. Postavke simulacije vatre i dima





*Slika 31. Render vatre i dima gdje dim zaobilazi prepreku*

## 6. ZAKLJUČAK

*Blender* je od jednog malog programa izrastao u jedan ozbiljan, profesionalni alat velikih mogućnosti dan na korištenje svima. U ovome radu se nije ulazilo preduboko u razne mogućnosti koje on posjeduje, ali se ipak moglo vidjeti što *Blender* ima za ponuditi. Od onoga što me najviše dojmilo osim simulacija je *engine* koji *Blender* koristi za renderiranje. *Engine* je zaista u stanju generirati impresivne vizualne prikaze, a to se posebno odnosi na prikaze svjetla i sjene koji jako upotpune scenu. Simulacijama fluida, vatre i dima mogu se postići zapanjujući vizualni prikazi i što je još važnije gotovo posve realni. Ja nažalost nisam uspio dobiti nešto na toj razini zbog prvenstveno tehničkih ograničenja, ali sam uspio pokazati ono što je bilo cilj ovog rada, a to je kretanje fluida u virtualnom prostoru i interakcija sa objektima. Iako se malo naprednijim snalaženjem u *Blender*-u i naprednijim poznavanjem postavki simulacija mogu dobiti realni prikazi sustav je daleko od gotovog proizvoda. Najveći problem kod simulacija je što se mora raditi doslovno na slijepo, jednostavno se ne može predvidjeti što treba podesiti u postavkama kako bi sigurno dobili ono što želimo. Svaki puta kada želimo provjeriti rezultate svojih novih podešavanja moramo renderirati cijelu simulaciju što je jako spor proces koji potrebno ponoviti mnoštvo puta kako bi smo dobili željenu simulaciju. Također računalni resursi koje je potrebno imati za ozbiljniji rad sa ovakvim simulacijama spadaju u sam vrh ponude danas. Netko tko posjeduje prosječno računalo današnjice se može, recimo to tako, igrati sa simulacijama jer jednostavno ta računala ne mogu podnijeti složenije simulacije. Meni je za *render* jednog *frame*-a simulacije vatre i dima trebalo više od 16 minuta, za jednu sliku, gdje simulacija fluida uopće nije bila uključena. Želio sam napraviti animaciju u kojoj se vide fluidi, vatra i dim zajedno, ali sam od toga mora odustati upravo zbog velikih zahtjeva za to. Sustav simulacije treba još mnogo optimizacije, ali je napredak vidljiv. Uz sve nedostatke fizikalni okvir simulacija *Mantaflow* je svakako jako moćan alat i nevjerojatno je da to sve možemo koristiti potpuno besplatno. Može se čak i reći da je malo ispred svog vremena te da će tek u skorijoj budućnosti, kada će naprednija tehnologija biti dostupnija svima, doći do punog izražaja.