

Jegyzőkönyv

KompGraf

2d grafika java nyelven

Inkscape vektorgrafika

Készítette: **Mihály Károly**

Neptunkód: **VECB52**

Dátum: 2024. május. 13

Tartalomjegyzék:

[KompGraf](#)

[2d grafika java nyelven](#)

[Inkscape vektorgrafika](#)

[Bevezetés:](#)

[1.Feladat](#)

[Java nyelvben egy 2d-s grafika elkészítése:](#)

[1.1 Konstruktor létrehozása](#)

[1.2 Importálások](#)

[1.3 Panel Képek létrehozása](#)

[1.4 Méret és elnevezés](#)

[1.5 Képek importálása](#)

[1.6 Panel szerkesztés előkészítése](#)

[1.7 Panel címe, színe](#)

[1.8 Első formák létrehozása](#)

[1.9 Képek beillesztése, szöveggel](#)

[1.10 Panel háttérszín és hozzáadás](#)

[2.Feladat.](#)

[Inkscape-ben elkészített grafika.](#)

[2.1 Első alakzatok](#)

[2.2 Színátmenetek](#)

[2.3 Alakzatok szerkesztése](#)

[2.4 Szöveg elkészítése](#)

[2.5 Szöveg árnyékolása](#)

[2.6 Szöveg útvonalra helyezése](#)

Bevezetés:

Feladatul kaptuk a KompGraf tantárgy befejezésére, hogy elkészítsünk 2 feladatot. Első feladatnak, egy java nyelven elkészített 2d-s grafika létrehozása. Én azt terveztem, hogy egyszerű grafikai elemekkel, és formákkal létrehozok egy kis ablakot, a JFrame segítségével. Úgy képzeltem el a projektet, hogy 4 képet elhelyezek az ablak 4 sarkába, és megszerkesztem őket, hogy kör formájuk legyen. Ezekre a képekre, keresztbe elhelyezek egy szöveget. Ez volt a legnehezebb része, a megoldásához nagyon sokat kellett kutatnom az interneten, de végül meg tudtam oldani, amit majd részletezek a feladat pontosabb leírásában. A képek után egy "X"-et terveztem pirossal az ablakba, ami a képek mögött haladt el, majd középre egy kék oválist, amibe egy "Sárospatak" szöveg áll. A képek köré még terveztem egy kék lekerekített téglalapot is.

Második beadandónak, egy inkscape projektre gondoltam, hogy egy vektorgrafikus képet készítek, a tanult metodikákkal. A projektben egy logo-t csináltam, ami tartalmaz 3 formát, színátmenetet, útvonalak közötti interpolálást, és szöveg elhelyezést útvonalra.

A mappaszerkezetem az végtelenül egyszerű ebben a projektben, ugyanis csak egy java projektet hoztam létre, ő felállította a futáshoz szükséges mapparendszert, és a programom mellé közvetlenül helyeztem el a felhasznált képeimet.

1.Feladat

Java nyelvben egy 2d-s grafika elkészítése:

1.1 Konstruktor létrehozása

A feladat elkezdéséhez létrehoztam egy új projektet, és első lépésem származtatni a JFrame osztályból és létrehozni a konstruktort a frame-hez. Ezután a main függvénybe megírni a hívását. Ez az alábbi módon néz ki:

```
public class KompGrafVECB52_1 extends JFrame{

    KompGrafVECB52_1() {
        . . .
    }

    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> {
            KompGrafVECB52_1 frame = new KompGrafVECB52_1();
            frame.setVisible(true);
        });
    }
}
```

1.2 Importálások

Miután sikerült a keretemet felhúzni a munkához, hozzáadtam a számomra fontos importokat. Az utolsó kettő érdekesnek tűnhet, de ahol használom, ott majd kitérek rá, hogy miért volt ez nekem fontos. Ezek az alábbiak voltak:

```
import javax.swing.*;
import java.awt.*;
import java.awt.geom.AffineTransform;
import java.awt.geom.Ellipse2D;
```

1.3 Panel Képek létrehozása

Mivel tudtam hogy 4 képet szeretnék, ezért az osztályon belül, de a konstruktoron kívül, létrehoztam őket. Ugyan itt, létrehoztam a panelt is, amire kerül majd a grafika.

```
private JPanel panel;
```

```
private Image image1;
private Image image2;
private Image image3;
private Image image4;
```

1.4 Méret és elnevezés

Ezek után, tudtam hogy mekkora ablakot szeretném, Így beállítottam az ablakom méretét, címét, és hogy bezáródjon az ablakom, hogy ha megnyomom az "X"-et. Ezt már a konstruktorba végeztem el.

```
super("VECB52 Java beadandó");
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setSize(500, 500);
```

1.5 Képek importálása

Ezek után, A képeknek adtam meg az elérési útvonalat, majd egy változóba eltárolom ezeknek az értékét, az alábbi módon, mind a 4 képpel:

```
ImageIcon icon1 = new ImageIcon("../\\..png");
image1 = icon1.getImage();
```

1.6 Panel szerkesztés előkészítése

Miután megvoltak az előkészületek, a létrehozott panel-emet, elkezdtem szerkeszteni. Ehhez szintén elő kellett állítani a környezetet a konstruktor-on belül. Ez a következőképpen néz ki a kódban:

```
panel = new JPanel() {
    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);
        Graphics2D g2d = (Graphics2D) g;
        /* Ide lehet rakni amiket meg akarunk jeleníteni, a g.valami() és
        a g2d.valami() segítségével */
    }
}
```

1.7 Panel címe, színe

Ezzel készen is voltam, így elkezdtem a panelem tartalmán dolgozni. Első dolgom volt, egy úgynevezett címet beállítani, ami a programom paneljének felső sorában jelenik meg. Ezt a következőképpen valósítottam meg.

```
g.setColor(Color.blue);
Font font = new Font("Arial", Font.BOLD, 24);
```

```
g.setFont(font);  
g.drawString("Mihály Károly beadandó", 100, 50);
```

1.8 Első formák létrehozása

Ezek után létrehoztam középre a tervezett oválisomat, majd a tervezett átlókat a programomban. Mivel az előbb kékre beállítottam a színt, ezért az ovális is kékkel rajzolódik ki, viszont a vonalaknak már piros színt állítottam, és egy 5 pixeles vastagságot. Ugyanígy még a leendő képek köré, egy kék kerekített sarkú téglalapot is helyeztem kék színnel, és a szöveget is hozzáadtam, fekete színnel.

```
g2d.fillOval(180, 190, 150, 120);  
  
g2d.setColor(Color.red);  
g2d.setStroke(new BasicStroke(5));  
g2d.drawLine(0, 0, 500, 500);  
g2d.drawLine(0, 500, 500, 0);  
  
g2d.setColor(Color.blue);  
g2d.drawRoundRect(25, 60, 450, 375, 20, 20);  
  
g.setColor(Color.black);  
g.drawString("Sárospatak", 200, 250);
```

1.9 Képek beillesztése, szöveggel

Most következik, a képek lekerekítése, és a rajtuk elhelyezkedő szöveg megszerkesztése. Elsőnek a a setClip metódust használtam a lekerekítéshez, mert neki meg lehet adni paraméterben egy elipszist, és a benne megjelenített kép, csak ebben a formában lesz látható, pont ahogy elterveztem. Ezután bele is illeszttem a clip-be a képet, az alábbi kódban, jelenleg a bal felső kép kódolása látható. Ezután az oldTransform nevű változóba, eltárolom a jelenlegi transzformációját a g2d objektumnak. Erre azért lesz szükség, hogy ha később átszerkesztjük, akkor vissza tudjuk állítani. a rotate függvény segítségével, elforgatom 30 fokkal, a kép középpontja körül. A szöveg kap egy piros színt, majd létrehozom egyszerű String változóként a szöveget, ami a képre kerül. Ezután jön képbe a másik könyvtár amit használok, a Metrics. Ennek segítségével ki tudom számolni egyszerűen. A metrics1 nevű változóban, letárolom a szövegem méretét. Miután letároltam, a függvény segítségével kiszámolom az x és az y koordinátát, ahova közép pontosítva lesz a szövegem elhelyezve. Ezután egyszerűen kirajzolom a szöveget a kiszámolt koordinátákra, majd visszaállítom a g2d transzformációját.

```
g2d.setClip(new Ellipse2D.Double(50, 100, 100, 100));  
g2d.drawImage(image1, 50, 100, 100, 100, this);  
AffineTransform oldTransform = g2d.getTransform();  
g2d.rotate(Math.toRadians(30), 50 + 50, 100 + 50);  
g2d.setColor(Color.red);  
String text1 = "LOGO";
```

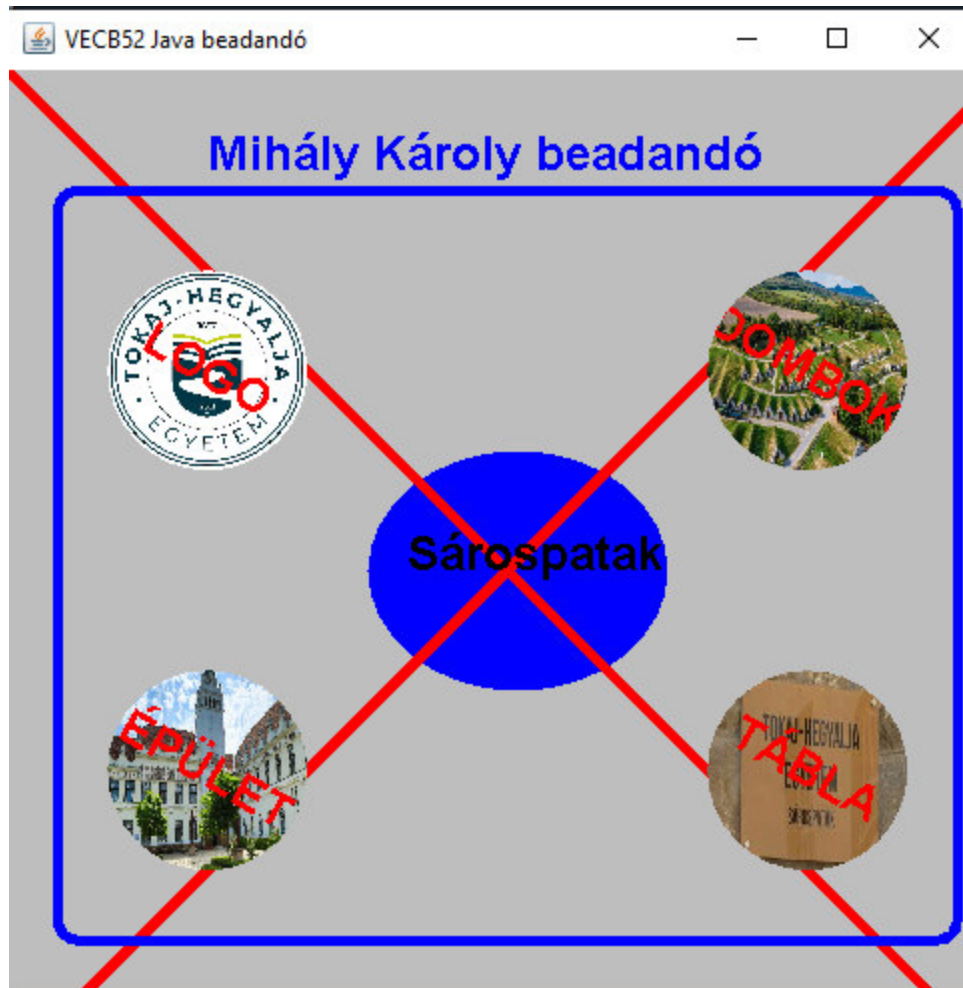
```
FontMetrics metrics1 = g2d.getFontMetrics();
int x1 = (100 - metrics1.stringWidth(text1)) / 2;
int y1 = ((100 - metrics1.getHeight()) / 2) + metrics1.getAscent();
g2d.drawString(text1, 50 + x1, 100 + y1);
g2d.setTransform(oldTransform);
g2d.setTransform(new AffineTransform());
```

1.10 Panel háttérszín és hozzáadás

A fenti kódot még megismételtem még 4 szer, mind a 4 képre, csak más koordinátákkal, és képekkel. Ez volt az utolsó lépés, amire szükségem volt a panelemben. Miután elkészültem, nem volt más dolgom, mint a panelnek egy egyszerű háttérszínt beállítani, majd a frame-hez hozzáadni a panelemet.

```
panel.setBackground(Color.LIGHT_GRAY);
add(panel);
```

Ezekkel a sorokkal, el is készült a programom. A futás, az alábbi módon néz ki:

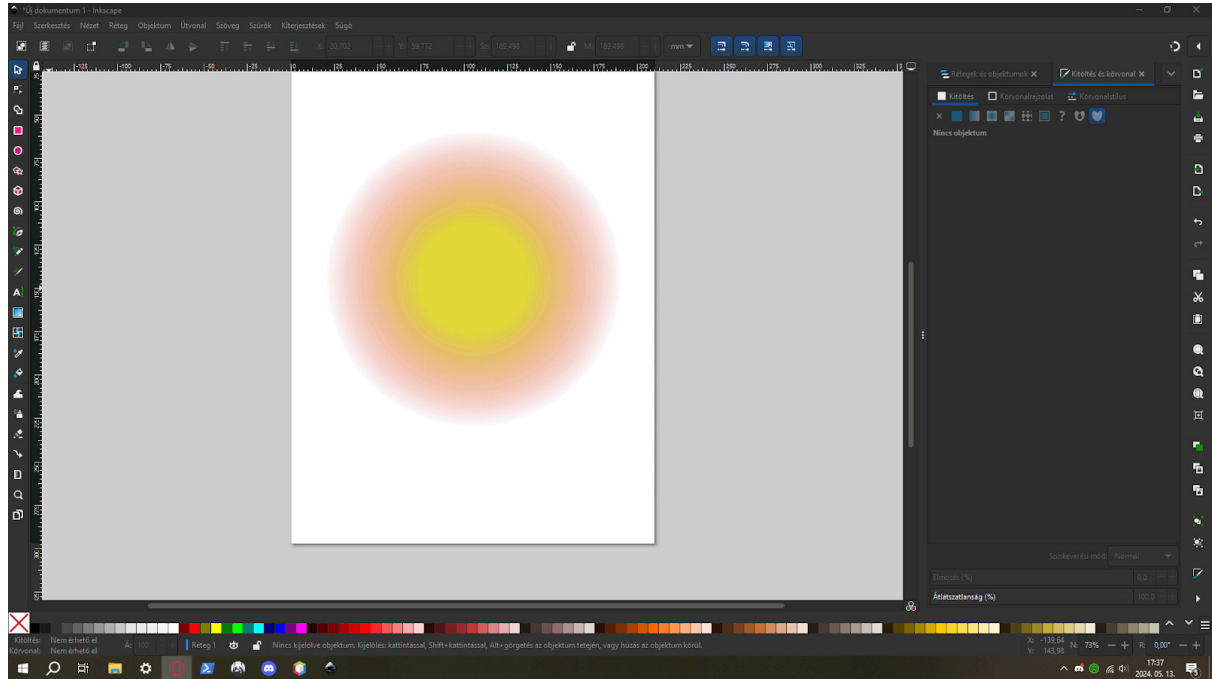


2.Feladat.

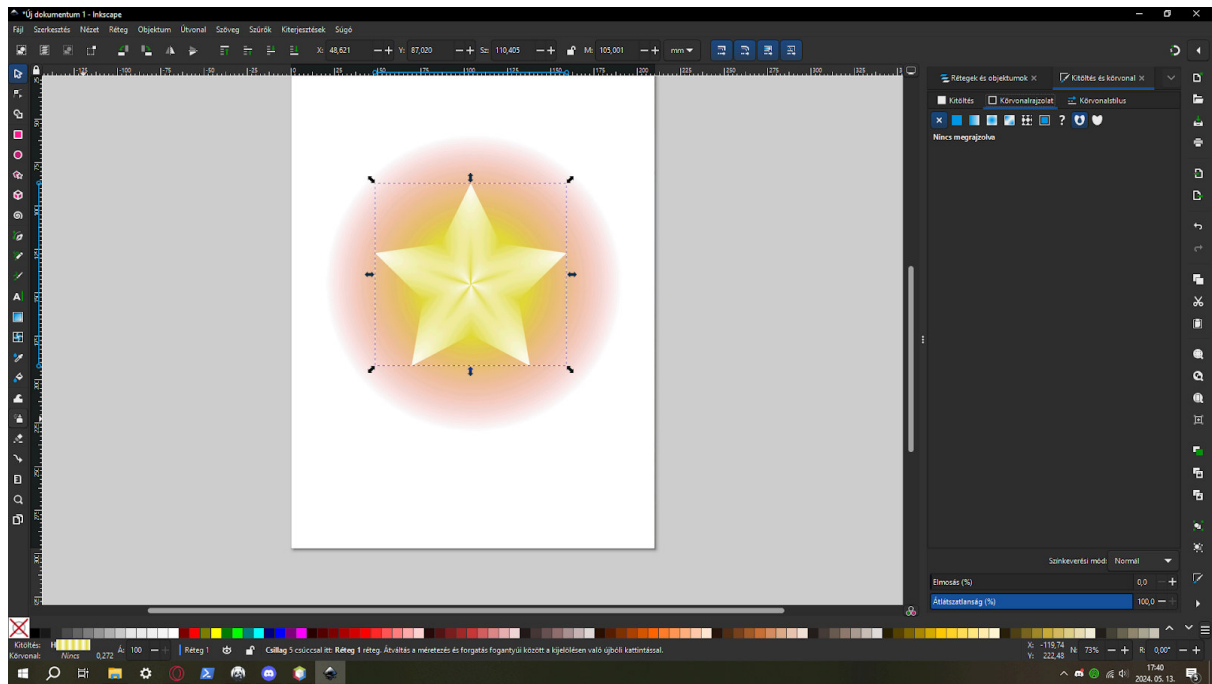
Inkscape-ben elkészített grafika.

2.1 Első alakzatok

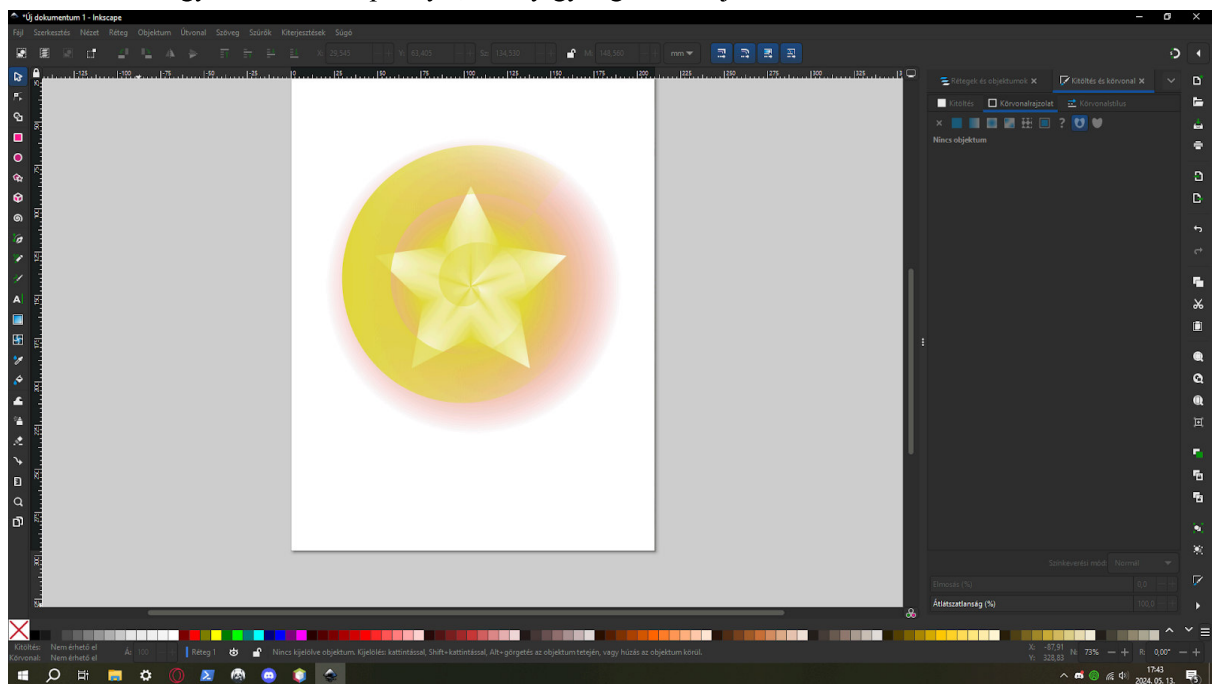
Első lépésnek, létrehoztam egy kör alakzatot, a projekt közepére. Erről a körrel leszedtem a körvonalat, és színátmenetet állítottam be neki.



Ezután a kör közepére elhelyeztem egy csillag alakzatot, amelyre árnyékolt kitöltést állítottam be. Ez a következő eredményezte:

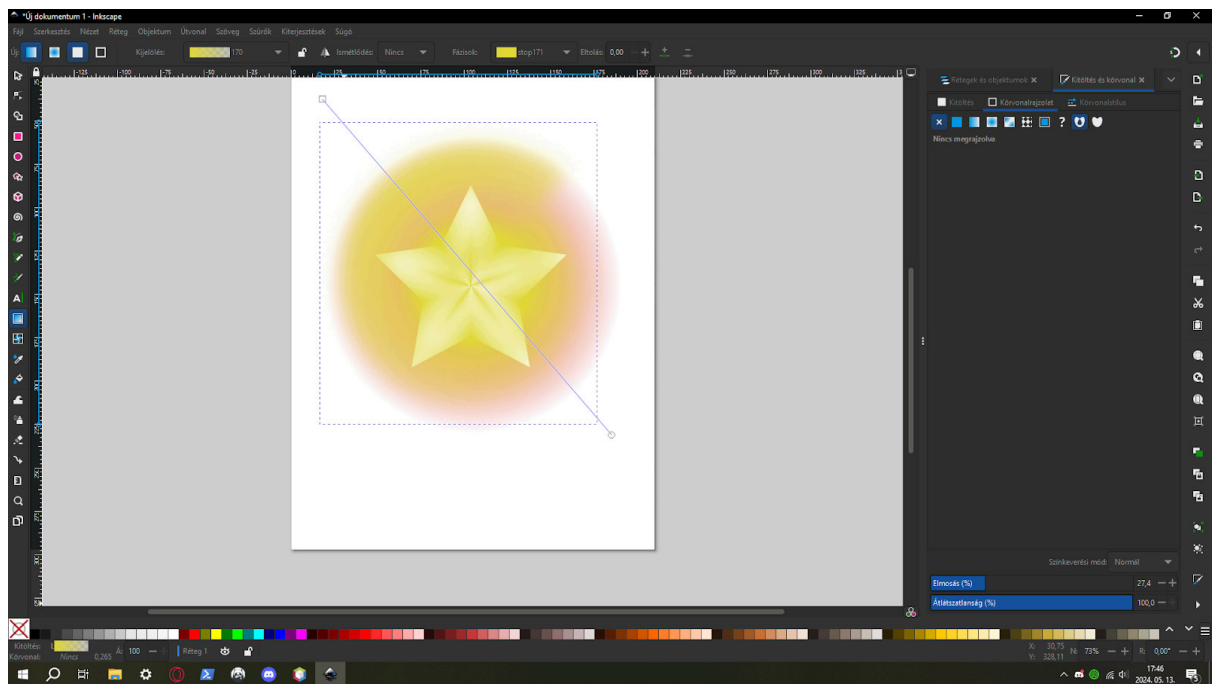


Miután formáztam a színeit, és az átmeneteit a csillagnak, létrehoztam egy spirál alakzatot ezekre, amire raktam egy 50% körüli opacity-t, amely gyengül a test jobb oldala felé.

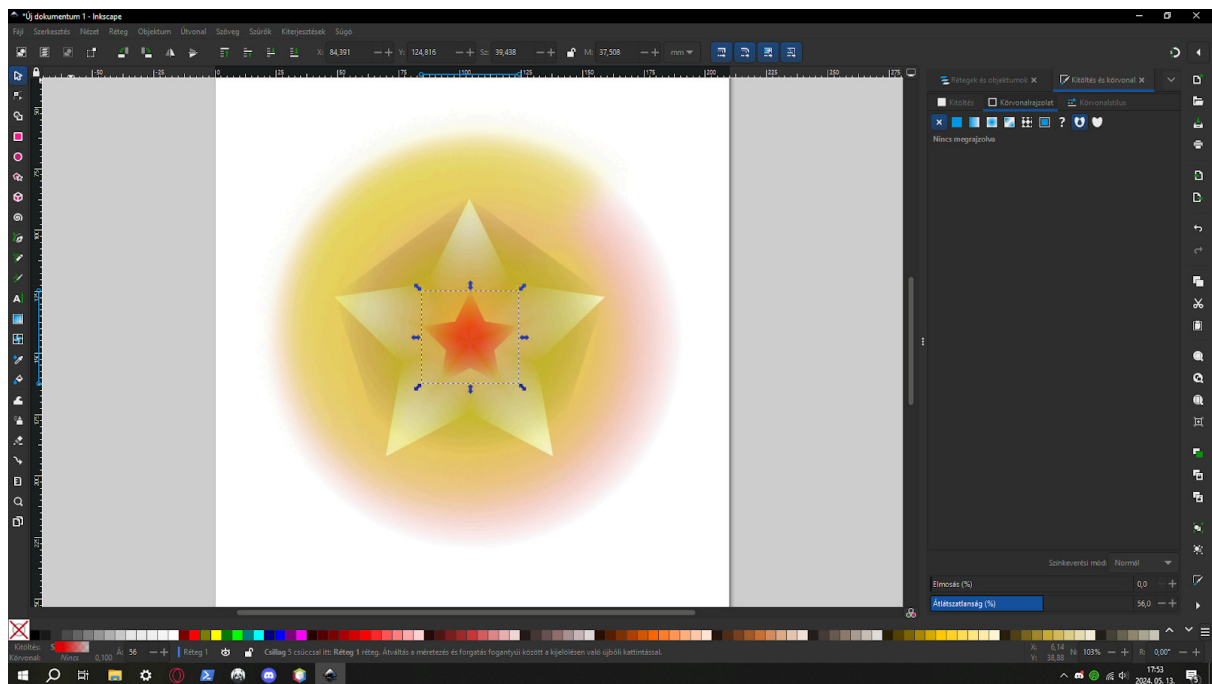


Viszont így túlságosan erős volt a spirál alakzata, ezért a fókuszpontjait átállítottam, ezzel elérve, egy elhomályosító hatást.

2.2 Színátmenetek

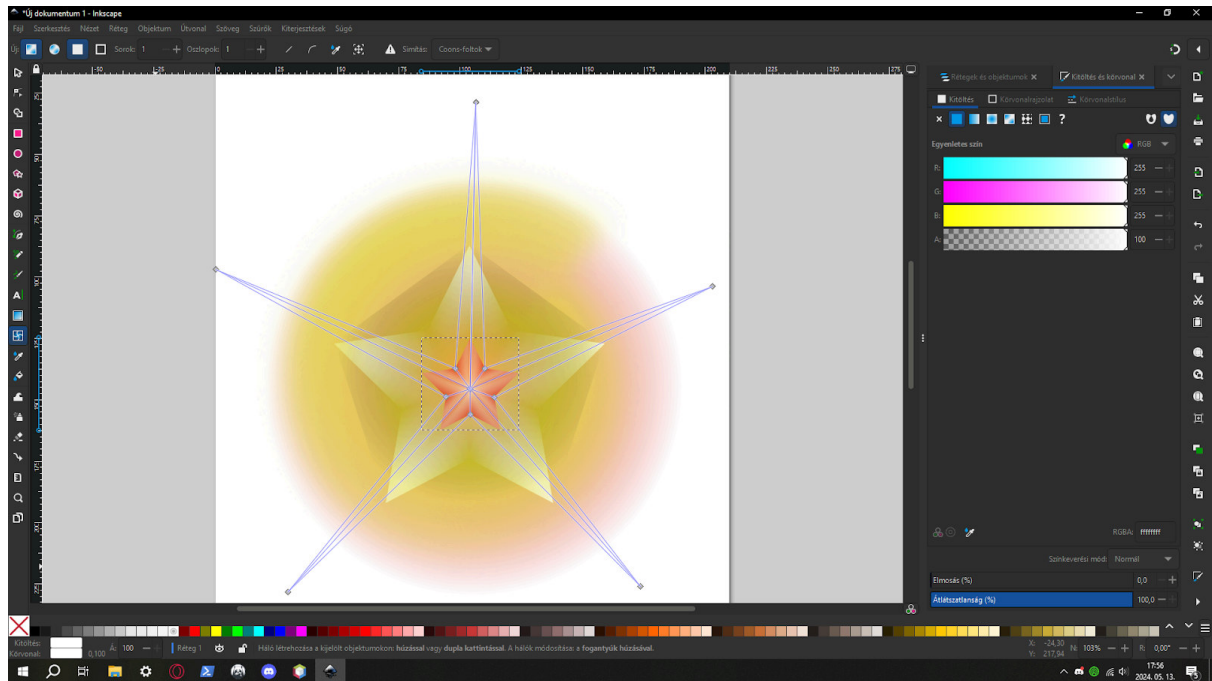


Miután elkészültem ezzel, szerettem volna a nagy csillagon belülre egy kisebb erősebb kitűnős színű csillagot elhelyezni, ezt meg is tettem.

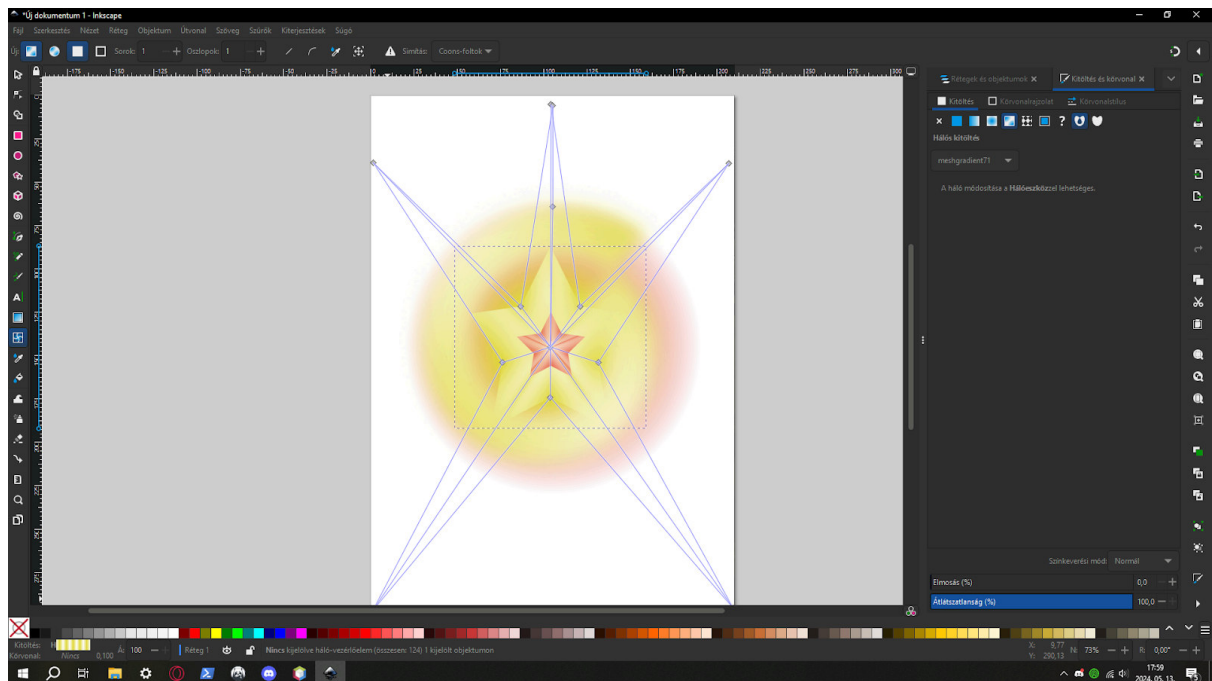


De mivel így nem igazán volt szép, ezért a csúcsainak az eltartását, beállítottam, az alakzaton kívülre, ezzel eredményezve az alábbi kinézetet.

2.3 Alakzatok szerkesztése



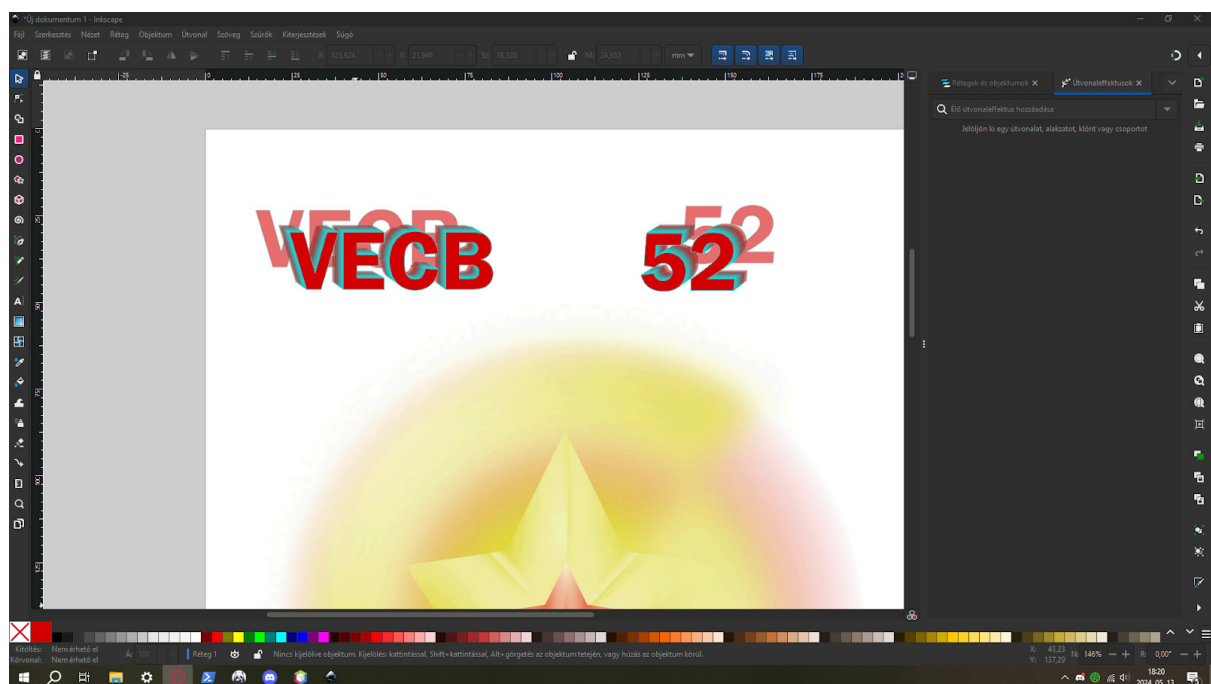
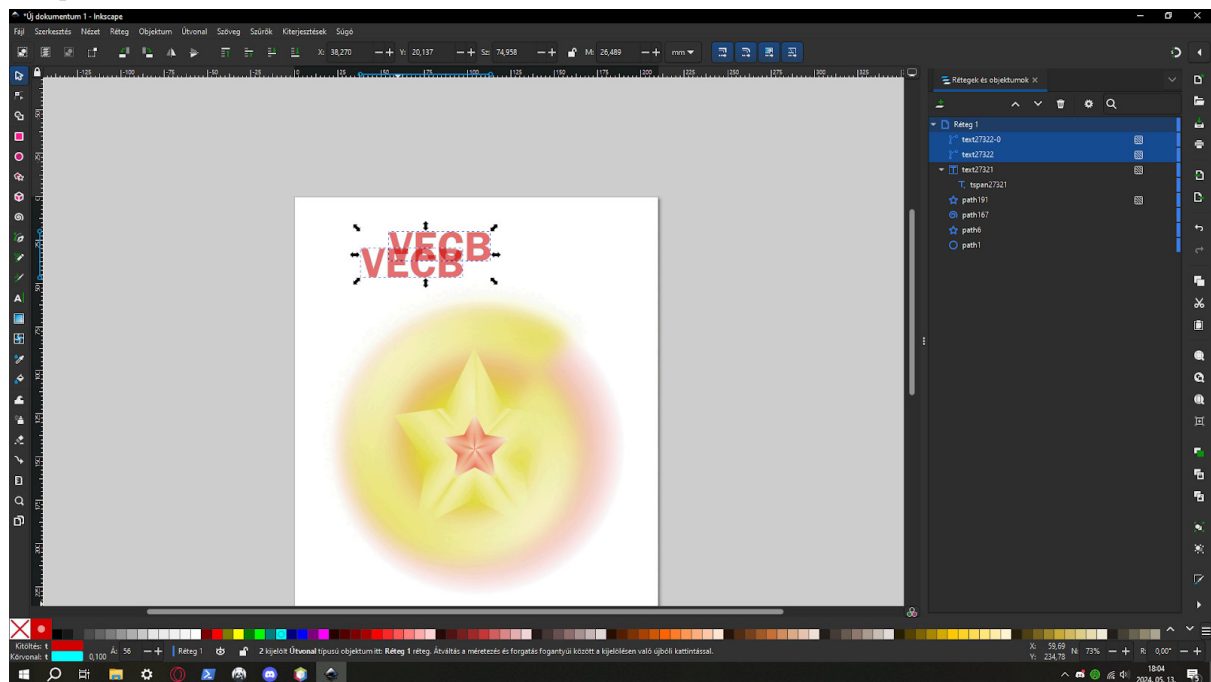
Hasonló procedúrát végeztem még a nagyobb csillaggal is.



2.4 Szöveg elkészítése

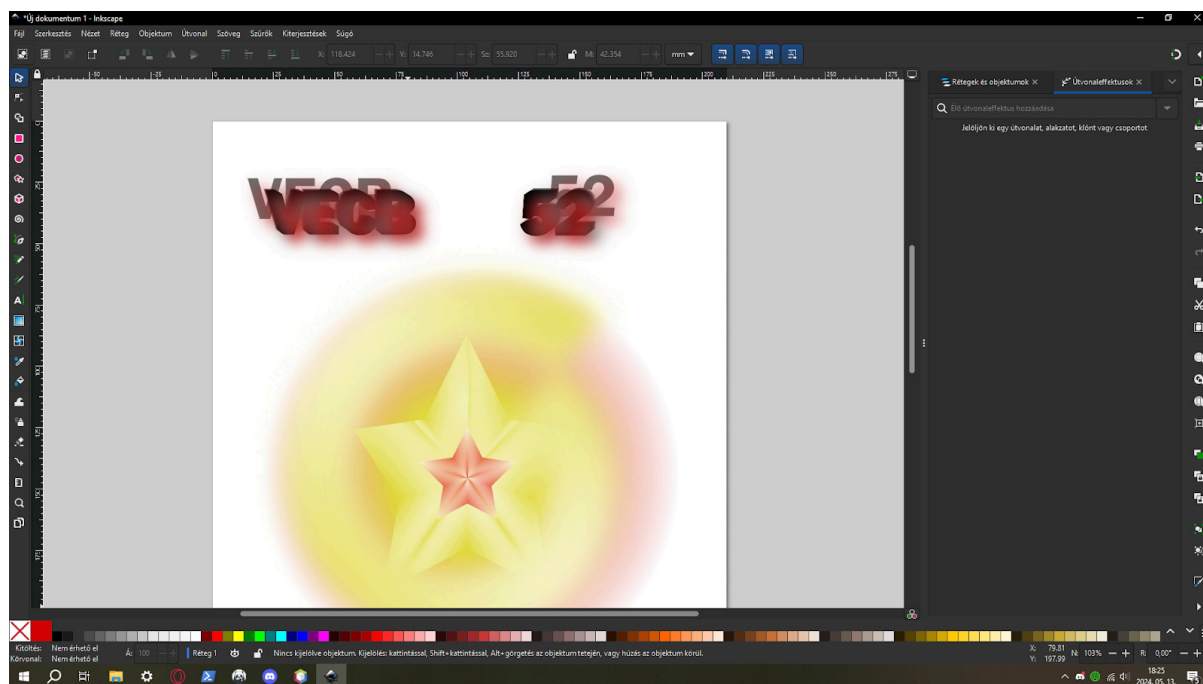
Miután elkészültem az alakzattal, elkezdtem dolgozni egy 3d-s szövegen, amit interpoláció segítségével tudtam megoldani. Szövegnek a neptun kódot választottam. Kiválasztottam a megfelelő szint és stílust, majd beírtam a szöveget. Ezután ezt a szöveg objektumot, útvonallá kellett

alakítanom, és másolni belőle még egyet, Ezzel el tudtam érni, hogy a 2 szöveg útvonal között, tudjak interpolálni, ami a 3d-s kinézet illusztrálását készíti el.



Miután elkészült a szöveg, csináltam neki egy árnyékolást, és fényezés effectet. Miután ezeket sikerült beállítanom és elkészítenem, az alábbi módon nézett ki.

2.5 Szöveg árnyékolása



2.6 Szöveg útvonalra helyezése

Miután készen volt ez, már csak egy szöveget készítettem, amit ráhelyeztem az alakzatom körvonalára.

