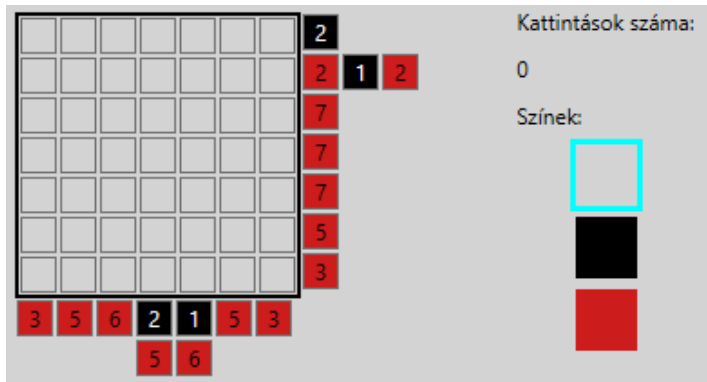


ColorCross Dokumentáció

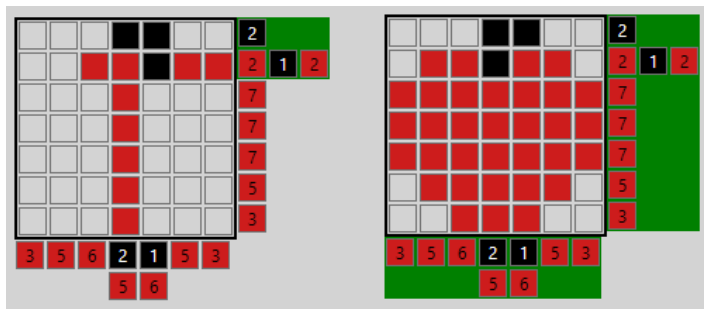
A Colorcross^{BCSZ} egy logikán alapuló „pixel” színező. A játék célja, hogy a pixeleket kiszínezve megkapjuk a pályán elrejtett képet. A pixelek helyes színezését a rajzterület mellett lévő táblázat adatai teszik lehetővé. Ebben megtalálható, hogy melyik sorban, illetve oszlopban összefüggően hány darab egyszínű pixel van egymás mellett. A számokat balról jobbra, illetve fentről lefelé olvasva kell figyelembe venni. Az átlátszó pixelek számát nem mutatja a táblázat (az alap képek esetén), mivel a képek háttere is átlátszóan van kitöltve, így ezen adattal túl egyszerű lenne megfejteni a képeket. A képek mellett a felhasználható színek listáját találjuk meg. Innen tudjuk kiválasztani, hogy milyen színnel szeretnénk színezni. A pixelek akárhányszor újraszínezhetőek, így akár a már helyesen kiszínezett pixelt is el tudjuk rontani egy rossz döntés esetén. A pálya akkor van kész, ha minden pixel helyesen van kiszínezve. A játékban látjuk, hogy hányszor kattintottunk már, így később megpróbálhatjuk újra kirakni a képet kevesebb lépésszámmal.

A játék kinézete:

A játék elején egy gombokkal teli ablakot kapunk, itt kell kirakni a képet, jobb oldalt egy lépésszámláló van, alatta pedig a felhasználható színek. Az első választható „szín” az átlátszó, így ha véletlenül olyan mezőt is kiszínezünk, amit nem kéne, akkor könnyen javíthatjuk a képet. A színeket kattintással lehet kiválasztani. Az éppen kiválasztott szín körül kék keret van.

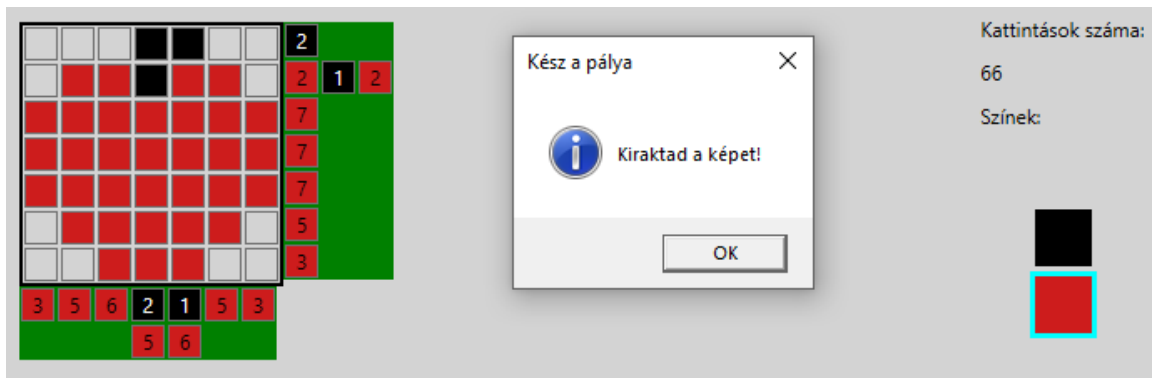


A játék zöldre színezi azon oszlopok/sorok táblázatrészeit, melyek megfelelő mennyiségben és sorrendben tartalmazzák a színeket. Ez nem jelenti azt, hogy jó is az adott sor/oszlop. Erre látunk példát a következő képen. Bal oldalt látjuk, hogy a felső két sor be van ugyan fejezve, de nem jó helyen van a kép szempontjából. Jobb oldalt látható a teljesen kirakott kép.

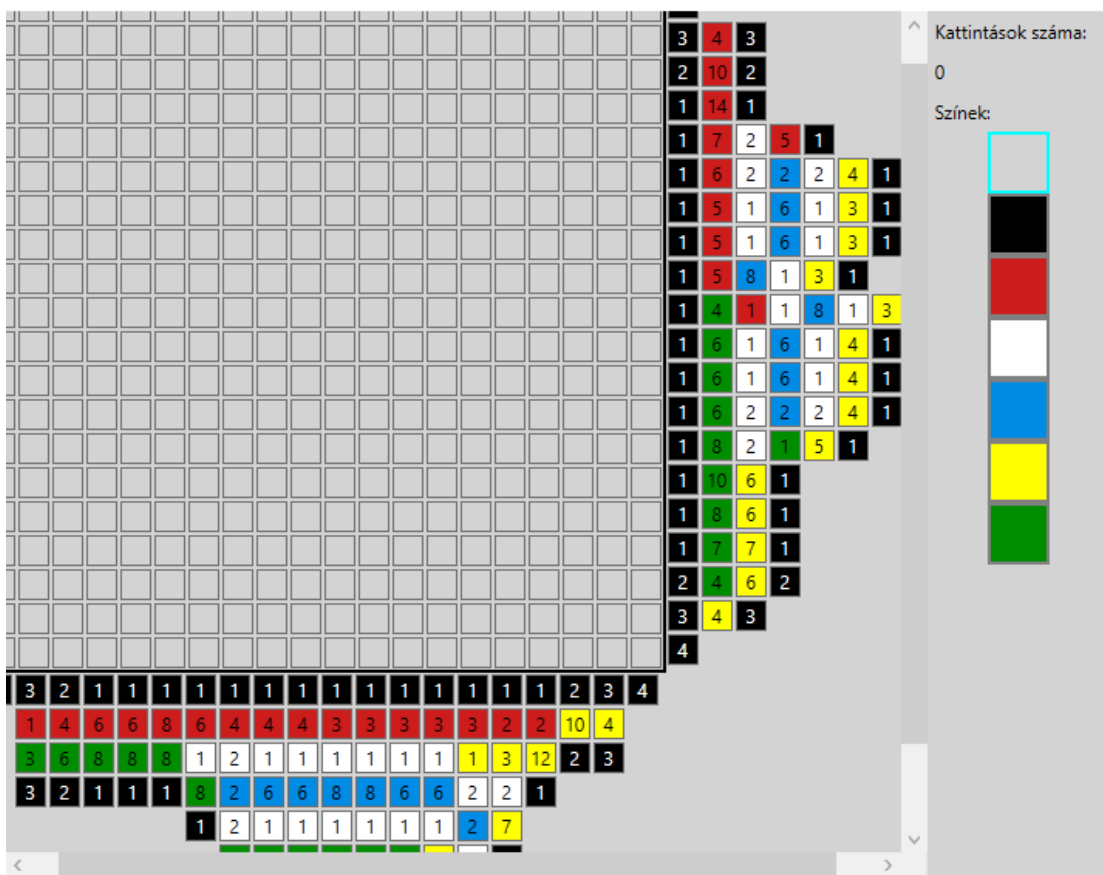


Ha sikerül kirakni a képet, akkor felugrik egy ablak, mely ezt közli is velünk. Ezt becsukva visszajutunk a főmenübe.

Ha még nem fejeztük be a képet és csak becsukjuk az ablakot, akkor is a főmenübe jutunk és a kép állása automatikusan elmentődik.



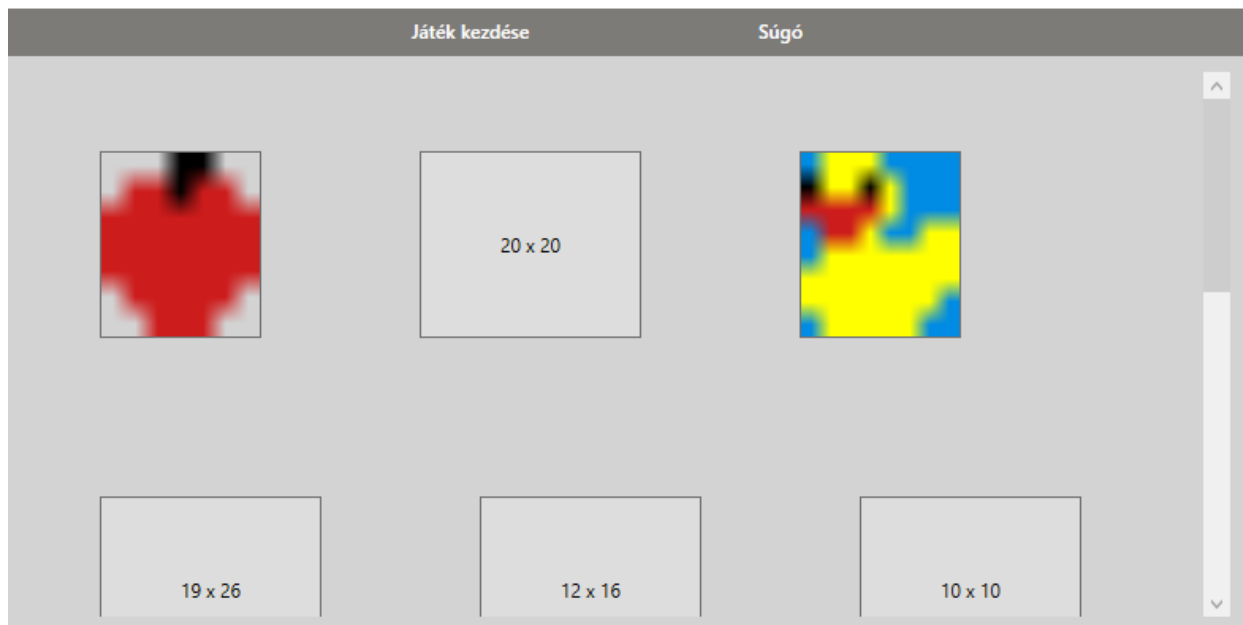
Ha túl nagyok a képek, akkor görgetősáv jeleni meg a kép alatt/mellett, így az összes kockát meg tudjuk nézni és ki tudjuk tölteni.



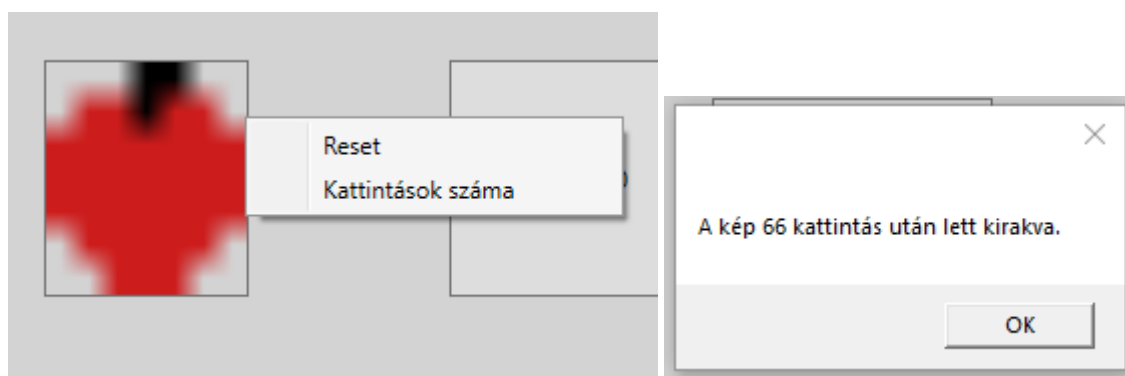
A főmenünek 2 oldala van, az egyik, ahol a képeket találjuk, a másik pedig a súgó.



Főoldalon a játék kezdése fülön találjuk a képeket. A még nem teljesített képek helyén csak a kép mérete látszik. A képre kattintva elkezdhetjük a pályát, vagy ha már egyszer elkezdtük, akkor a mentett állapottól folytatjuk.



A kész képek viszont már megjelennek a pályák helyén. A képre jobb gombbal kattintva 2 választási lehetőség kerül elő. Megnézhetjük a megoldás alatt felhasznált lépések számát, vagy visszaállíthatjuk a pályát az eredeti, üres állapotára. Így bármikor újracsinálhatjuk a pályákat.



A képek bmp formátumban vannak, így a mappához bármikor adhatunk új képet ezzel a játékot bővítve.

Tapasztalatok a csapatmunka alatt:

A feladat nehézségei ott kezdődtek, hogy 3 „jó lesz az utolsó pillanatban” és „nem tudom, amit te szeretnél” döntésképtelen ember nehezen kezd hozzá bármihez. Majd ott folytatódtak, hogy kreativitás kellett a feladathoz, a specifikációhoz és most a dokumentációhoz. A kezdeti produktivitás mértékén természetesen az se segített, hogy

hetek kellettek ahhoz, hogy egyáltalán oda eljussunk, hogy mit szeretnénk csinálni. Annyira megőrültünk, amikor végre kitaláltuk a játék típusát, hogy itt abba is hagytuk a tárgyalást a feladatról és egy újabb hét kellett hozzá, hogy a nagyon vázlatos kinézetet megálmodjuk. Aztán eljött a specifikáció leadásának határideje, úgyhogy muszáj volt valamit produkálnunk. Itt jöttünk rá, hogy nem biztos, hogy a legjobb feladatot választottuk, mert a követelményeket elolvasva felfedeztük, hogy kéne egy ranglista is, ami egy színezőnél szokatlan dolog. Új játékot találni már nem volt idő, így a kreativitás nyert és kitaláltuk, hogy majd stopperrel mérjük a kirakási időt. Ez elméletben ugyan jó ötletnek tűnt, de mint utólag kiderült gyakorlatban nem tudtuk úgy megvalósítani, ahogy szeretnénk volna, így el kellett vetni ezt a megoldást. Visszatérve a specifikációhoz, a határidő végre meghozta a motivációt, és amit hetek alatt nem sikerült megbeszelnünk végül kész lett egy nap alatt. Igaz még itt se volt rendesen felépítve a program vagy kitalálva a logikai alap, de legalább a fő funkciókról és az ablakok kinézetéről lett egy halvány képünk. A leadás után még véletlenül se éreztük úgy, hogy jobban bele kéne ásnunk magunkat, hiszen a specifikáció is kész lett időre. Nem kell aggódni, nem lesz baj, ugye? Jó, jó, azért el kéne kezdeni kódolni is.... Itt jött a kérdés, hogy, de mit is kéne ténylegesen megvalósítani, hogyan osszuk el a feladatokat, ha az sincs meg, hogy amúgy mik a feladatok? Nem baj, nincs konkrét vázlat, nem is kell, majd kialakul. Bence csinálja a frontendet, Csongor a backendet, Zita meg a kettő közötti dolgokat és a kis plusz feladatokat és szépen lassan folyamatosan haladunk. Legalábbis ez volt a nagy haditerv, amit természetesen véletlenül se követtünk. Volt itt minden, sértődések, könyörgések, hogy a másik tegye fel a kódot, amivel már készen van, „elfelejtettem, hogy létezik a specifikáció”, egymás kódjának az értelmezése, „az miért ott van?”, „az miért nem úgy van?”, „miért intet ad vissza?”. Teljes katasztrófa és természetesen rengeteg halogatás. Oh és ezek után se tanultunk a hibánkból és még véletlenül se álltunk meg tervezni (mondanám, hogy újratevezni, de nem volt mit).

Úgyhogy itt vagyunk az utolsó napokban a káosz közepén, hajnali kettőkor még a kódot javítgatva és dokumentációt írva... Mert ugye ráérünk még, ugye?

Utolsó nap és még most írjuk át a rétegsértéseket, hogy kell a gombokat dinamikusan felrakni? Na és a kiírás? Az nem jó, összezsúszik az egész. Kell még a mentés is. Itt ülünk már a sokadik kávéval és mindenki egyszerre csinál mindent. Ilyenkor jön elő a „Jaj, a mastert le kéne szedni gitről, de még nincs meg a merge” vagy a „senki nem ír most a logicba, ugye? Akkor legyünk bátrak, megy rögtön a masterbe és le tudod szedni a javítást.”

Amikor a google 3. oldalán se találsz megoldást és Stack Overflow is cserben hagy, akkor lehet elgondolkozol rajta, hogy igen, ha hamarabb elkezdtem volna.... De most már kár ezen gondolkodni, majd legközelebb időben nekiállok (és egyéb ígéretek, amiket úgyse tartok be). Két igen komoly motiváció tartja bennünk a lelket, az egyik a határidő (~~és, hogy nem szeretnénk fizetni~~), a másik, hogy amint végzünk, be lehet csukni az összes fölös ablakot, mint például ezeket.



Mit ne mondjunk, a kódunk pont olyan nagy káosz volt, mint ez a dokumentáció is. De a káoszban születnek a jó ötletek (néha). Lehet nem a legszebb és a legoptimálisabb a játékunk szerkezete vagy nem a legbonyolultabb a játék, de működik. Volt, hogy refaktorálni kellett a fél logicot, mert kiderült, hogy egy-egy funkció csak úgy működik megfelelően. Volt, hogy ki kellett hagyni egy megálmodott részt. Sok elakadás, pár vita, egymás kijavítása, és néhány siker után azért sikerült egy működőképes játékot összehoznunk és nem fojtottuk meg egymást közbe, ami valljuk be, nagy teljesítmény.

A következőkben szeretnénk felsorolni, hogy mik mentek nehezebben, mikkel voltak gondjaink, illetve, hogy ne csak a rossz dolgok kerüljenek előtérbe, azt is szeretnénk kifejezni, hogy mikre vagyunk büszkéek és bizonyos nehezebb problémákat hogyan sikerült megoldanunk.

Első nagyobb probléma még mindig a döntésképtelenség és halogatás, de ezt a határidők megoldották.

Nehézséget okozott, hogy nem volt rendesen megtervezve a program alapja és ez a fejetlenségben meg is látszott, hiszen nem volt hol elkezdni az egészet.

Problémás volt a színeket tartalmazó táblázat beforgatása és megfelelő tagolása, amit végül plusz griddel és StackPanellel sokadik próbálkozásra, de sikerült megoldani és ez talán nem tűnik nagydolognak, de én nagyon örültem neki, hogy végre sikerült.

A tervezés hiányából adódóan sokszor újra kellett írni a kódot, egymás kódját változtatni és úgy megérteni a másik kódját, hogy nem tudtam mit szeretne vele.

A rétegsértések kijavítása és a több rétegen keresztül való adatátvitel is sok fejfájást okozott, főleg amikor nem tudtam, hogy melyik rétegből kéne a gombok felrakását és azok színváltoztatását megoldani. Később derült ki, hogy nincs meg és ami van belőle az is rossz rétegben van, így kicsit bátortalanul kezdtem hozzá, hiszen más kódját nem akartam átírni, de Stack Overflowon egy viszonylag jó alapot találtam és meg lett oldva ez is.

Ami nagyon nagy hátrány volt, hogy egyikünk se szokott meg, hogy csapatban kell dolgozni és ezért eléggé elmaradtak a gitre a pusholások és a mergelek, azán csodálkoztunk, hogy van, akinek megy a kód és van, akinek nem.

Nehézséget okozott a specifikációban 3 oldalt írni (ami végül sok képpel, de több is lett), illetve a dokumentáció megírása, mert egyszerűen nem tudok/tudunk ennyit írni, illetve még nincs sok tapasztalatunk ilyen dokumentumok írásában.

Amire büszke tudok lenni, azok a konverterek és például, hogy a játék kijelöli, hogy melyik szín van használatban, kis apróság, de rengeteg hibát sikerült csinálni benne és hihetetlen, hogy működik.

Az első igazi nagy nehézséget az jelentette (azután, hogy mi legyen a játék témája), hogy mégis hogyan fog a játék működni kód szinten. Először jött a hirtelen ötlet, mivel, hogy a

játék színekből áll, ezért tároljunk, és kezeljünk mindent szín objektumokkal. Ez már a kezdetekben is fejfájásokat okozott, de csak azért se változtattunk. Végül a rengeteg átírás és a későbbi, esetleges problémák miatt átváltottunk „int” alapú színkezelésre, ami hatékonyabb is (sokkal) és áttekinthetőbb.

Számomra a legnagyobb problémát maga a kirajzolás jelentette. Elképzelésem se volt, hogy hogy kellene kirajzolni és interaktívvá tenni a felületet. Többszöri átnézés után kezdtem el kapizsgálni, hogy Zita, hogy oldotta meg az egészet, elképesztő. Mivel az egész játék azon az egy ablakon alapult, ezért túl sokat nem is tudtam haladni amíg az nem volt használható. Nagyon sok minden attól függött és vakon nem szerettem volna kódolni (így is sok volt belőle). Alapból se szeretem az egész frontend területet, WPF se nyerte el a tetszésem. A háttérlogika legtöbb részét internetet keresgélés és logikázás után szépen meg lehetett alkotni.

Ranglista ötleténél is problémák után jöttek a problémák. Először Stopwatch-csal próbálkoztunk. Sikerült is, működött, frissült is az ablakon az eltelt idő. Ott jött a probléma, hogy el is kéne menteni az állást, vagyis az aktuális időt. Mivel itt már szerializációs problémák is előjöttek, ezért ez el lett felejtve. Egyéb próbálkozások után végül a kattintások számánál maradtunk. Egyszerű számként lementjük, hogy hány kattintásból sikerült kirakni a képet. Ugyan rendes ranglista nincs, de a lelkesebbek között a minél kisebb számra való törekvés is létrehozhat versengést.

Kiemelni nem is tudnék olyan részt amire igazán büszke vagyok. Mindent sikerült kilogikázni és születtek szép megoldások, de ezeket nem gondolnám jelentősen kiemelkedőnek. Ha nagyon ki kéne emelni pár dolgot akkor elsőnek a kép ellenőrzését említeném meg. Nem csak annyi, hogy egyesével végig megyünk minden egyes cellán, ez hangzásra is lassúnak tűnik. Ehelyett egy gyorsabb megoldást találtam ki az ellenőrzésre, mindig a változtatott sort/oszlopot ellenőrzöm, hogy egyezik-e az eredeti színmintával. Végül, ha a jól kirakott sorok/oszlopok száma egyezik az összes sor/oszlop számával akkor biztosan kész a kép. Ehhez kapcsolódik az az algoritmus is, ami létrehozza a listát, melyben megtalálható, hogy az adott sorban vagy oszlopban melyik színből hányas méretű blokkok találhatóak.

Ezek mellett a jobb klikkes menüt is kiemelném. Ugyan nem volt vele se probléma, se nagy munka, de ilyenekkel még nem foglalkoztam és tetszett ahogy összeállt az egész. Alapból csak egy kényszer megoldás lett a kép visszaállítására és a kattintások számának kijelzésére, ugyanis nem nagyon tudtam, hogy hol máshol valósítsam meg ezeket.

Összességében sok nehézség elé kerültünk, de valahogy sikerült megcsinálni a játékot és kiderült, hogy igaz mind szeretünk halogatni, de viszonylag jól össze tudunk dolgozni, amikor már nyakunkon a leadás ideje.