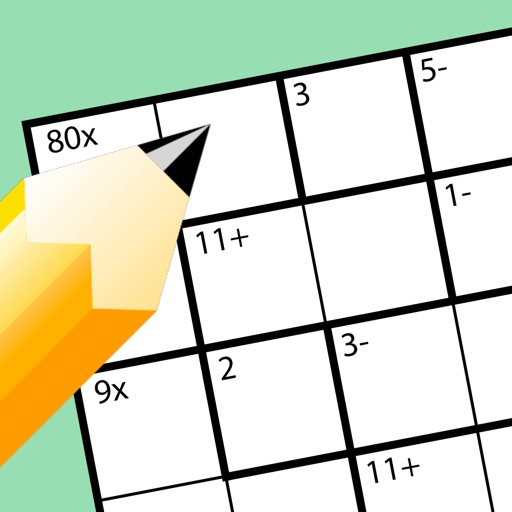
Mathdoku



Cuprins

[Tabel de figuri 4](#_Toc138160845)

[Capitolul 1. Introducere 6](#_Toc138160846)

[1.1 Obiectivele lucrării 6](#_Toc138160847)

[1.2 Motivația alegerii temei 7](#_Toc138160848)

[1.3 Contribuția personala 7](#_Toc138160849)

[Capitolul 2. Stadiul actual al puzzle-urilor matematice 8](#_Toc138160850)

[2.1 Originea și evoluția Mathdoku 8](#_Toc138160851)

[2.2 Regulile și variantele Mathdoku 10](#_Toc138160852)

[2.3 Beneficiile rezolvării de puzzle-uri Mathdoku 11](#_Toc138160853)

[Capitolul 3. Concepte, fundamente și tehonologii utilizate 13](#_Toc138160854)

[3.1 Fundamente matematice ale Mathdoku 13](#_Toc138160855)

[3.2 Tehnologii utilizate în dezvoltarea aplicației Mathdoku 14](#_Toc138160856)

[3.3 Utilizarea Rapid7 în dezvoltarea aplicației Mathdoku 15](#_Toc138160857)

[3.4 Utilizarea ALM în dezvoltarea aplicației Mathdoku 16](#_Toc138160858)

[Capitolul 4. Dezvoltarea aplicației Mathdoku 17](#_Toc138160859)

[4.1 Analiza aplicației Mathdoku 17](#_Toc138160860)

[4.2 Proiectarea aplicației Mathdoku 18](#_Toc138160861)

[4.3 Implementarea aplicației Mathdoku 22](#_Toc138160862)

[4.4 Testarea aplicației Mathdoku 44](#_Toc138160863)

[Capitolul 5. Concluzii 54](#_Toc138160864)

[5.1 Rezultatele obținute 54](#_Toc138160865)

[5.2 Concluzii generale 54](#_Toc138160866)

[Bibliografie 55](#_Toc138160867)

# Tabel de figuri

[Figure 1 19](#_Toc138160868)

[Figure 2 20](#_Toc138160869)

[Figure 3 21](#_Toc138160870)

[Figure 4 23](#_Toc138160871)

[Figure 5 23](#_Toc138160872)

[Figure 6 24](#_Toc138160873)

[Figure 7 24](#_Toc138160874)

[Figure 8 24](#_Toc138160875)

[Figure 9 25](#_Toc138160876)

[Figure 10 25](#_Toc138160877)

[Figure 11 25](#_Toc138160878)

[Figure 12 26](#_Toc138160879)

[Figure 13 27](#_Toc138160880)

[Figure 14 27](#_Toc138160881)

[Figure 15 28](#_Toc138160882)

[Figure 16 28](#_Toc138160883)

[Figure 17 29](#_Toc138160884)

[Figure 18 30](#_Toc138160885)

[Figure 19 30](#_Toc138160886)

[Figure 20 31](#_Toc138160887)

[Figure 21 32](#_Toc138160888)

[Figure 22 33](#_Toc138160889)

[Figure 23 34](#_Toc138160890)

[Figure 24 34](#_Toc138160891)

[Figure 25 35](#_Toc138160892)

[Figure 26 36](#_Toc138160893)

[Figure 27 37](#_Toc138160894)

[Figure 28 38](#_Toc138160895)

[Figure 29 39](#_Toc138160896)

[Figure 30 40](#_Toc138160897)

[Figure 31 41](#_Toc138160898)

[Figure 32 42](#_Toc138160899)

[Figure 33 43](#_Toc138160900)

[Figure 34 44](#_Toc138160901)

[Figure 35 45](#_Toc138160902)

[Figure 36 45](#_Toc138160903)

[Figure 37 46](#_Toc138160904)

[Figure 38 47](#_Toc138160905)

[Figure 39 47](#_Toc138160906)

[Figure 40 48](#_Toc138160907)

[Figure 41 48](#_Toc138160908)

[Figure 42 48](#_Toc138160909)

[Figure 43 49](#_Toc138160910)

[Figure 44 49](#_Toc138160911)

[Figure 45 50](#_Toc138160912)

[Figure 46 50](#_Toc138160913)

[Figure 47 51](#_Toc138160914)

[Figure 48 51](#_Toc138160915)

[Figure 49 52](#_Toc138160916)

[Figure 50 52](#_Toc138160917)

[Figure 51 53](#_Toc138160918)

[Figure 52 53](#_Toc138160919)

# Capitolul 1. Introducere

Mathdoku este un puzzle matematic inspirat din Sudoku care îți pune la încercare abilitățile logice și aritmetice. Mathdoku, cunoscut și sub numele de KenKen sau Calcudoku, este un joc de puzzle captivant și educativ, conceput pentru a te provoca și a-ți îmbunătăți competențele matematice.

Asemănător cu Sudoku, Mathdoku utilizează o grilă pătrată împărțită în mai multe regiuni mai mici, denumite “căsuțe” sau “cages”. În Mathdoku, trebuie să folosești combinații de cifre care satisfac anumite condiții aritmetice. Aceste condiții pot fi adunări, scăderi, înmulțiri sau împărțiri și sunt indicate de un simbol specific în fiecare căsuță.

Obiectivul tău în Mathdoku este să completezi grila cu cifre în așa fel încât să îndeplinească toate cerințele aritmetice pentru fiecare regiune. Aceasta înseamnă că într-o regiune, suma, diferența, produsul sau câtul dintre cifre trebuie să se potrivească cu rezultatul specificat. Cu cât rezolvi mai multe puzzle-uri Mathdoku, cu atât devii mai abil în rezolvarea provocărilor matematice complexe.

Mathdoku este un joc minunat pentru toți cei pasionați de matematică și puzzle-uri, oferindu-le o modalitate distractivă și stimulantă de a-și exersa gândirea logică și abilitățile matematice. Indiferent dacă ești un începător sau un expert în matematică, Mathdoku te va captiva și te va provoca constant, deschizându-ți oportunitatea de a descoperi și de a rezolva noi puzzle-uri matematice în fiecare zi.

## 1.1 Obiectivele lucrării

Lucrarea aceasta privind tema Mathdoku se va concentra pe prezentarea detaliată a puzzle-ului matematic, evaluarea beneficiilor sale și explorarea modalităților de utilizare în educație și divertisment. Scopul final este de a inspira și de a încuraja cititorii să exploreze lumea fascinantă a Mathdoku și să beneficieze de provocările matematice și de distracția pe care acesta le oferă.

## 1.2 Motivația alegerii temei

Matematica este o disciplină fundamentală și esențială în dezvoltarea noastră intelectuală. Ea ne ajută să înțelegem lumea înconjurătoare și să rezolvăm probleme complexe în viața de zi cu zi.

Mathdoku reprezintă o modalitate inovatoare și captivantă de a ne apropia de matematică prin intermediul unui joc interactiv. Alegerea acestui proiect se bazează pe dorința mea de a aduce matematica mai aproape de oameni și de a-i provoca să gândească logic și să-și dezvolte abilitățile matematice într-un mod distractiv și stimulant.

Mathdoku este o formă de puzzle care promovează gândirea critică și rezolvarea de probleme. Jucătorii trebuie să analizeze cu atenție cerințele aritmetice și să găsească combinațiile corecte de cifre pentru a completa grila. Această abordare îi încurajează pe oameni să-și exercite abilitățile de raționament și să găsească soluții creative la provocări matematice.

## 1.3 Contribuția personala

Contribuția personală în cadrul acestui proiect constă în cercetarea, documentarea și redactarea informațiilor relevante despre Mathdoku. Am petrecut timp pentru a studia originile și evoluția acestui puzzle matematic, regulile și variantele sale, precum și beneficiile pe care le poate aduce în dezvoltarea abilităților matematice și cognitive.

Am analizat studii și resurse educaționale pentru a obține o înțelegere aprofundată a impactului Mathdoku în contextul educației și al dezvoltării copiilor și tinerilor. Am cercetat, de asemenea, strategii de rezolvare și modele de puzzle-uri pentru a oferi informații practice și utile în rezolvarea acestui tip de puzzle.

În timpul redactării, am încercat să explic conceptele matematice și regulile Mathdoku într-un mod clar și accesibil, astfel încât oricine să poată înțelege și să se bucure de acest joc. Am urmărit să evidențiez beneficiile matematice și cognitive ale Mathdoku, subliniind importanța acestui puzzle în dezvoltarea gândirii critice și a abilităților de rezolvare a problemelor.

# Capitolul 2. Stadiul actual al puzzle-urilor matematice

În acest capitol, voi evidenția stadiul in care se afla puzzle-urile matematice in momentul de fata, modul in care puzzle-urile de acest tip au evoluat pe parcursul tipului si originea acestora, regulile jocului si beneficiile rezolvării acestui puzzle captivant.

Stadiul actual al puzzle-urilor matematice este unul dinamic și diversificat. Aceste puzzle-uri au devenit extrem de populare și au evoluat într-o multitudine de forme și variante, oferind o varietate de provocări matematice pentru toate vârstele și nivelurile de abilități.

Printre cele mai cunoscute și apreciate puzzle-uri matematice se numără Sudoku, Mathdoku, Kakuro, Puzzle-urile logice și multe altele. Acestea au câștigat popularitate în întreaga lume și sunt disponibile în diverse medii, inclusiv în formă de jocuri pe calculator, aplicații mobile, cărți de puzzle-uri și chiar în formă fizică.

Un aspect notabil al stadiului actual al puzzle-urilor matematice este adaptabilitatea lor la mediul digital. Dezvoltarea tehnologiei și accesibilitatea dispozitivelor inteligente au permis puzzle-urilor matematice să se extindă și să fie disponibile online sau prin intermediul aplicațiilor mobile. Aceasta a oferit oportunitatea de a explora și de a experimenta puzzle-uri matematice într-un mod interactiv, ușor de accesat și de partajat cu alții.

Un alt aspect important este faptul că puzzle-urile matematice nu sunt doar o formă de divertisment, ci și o modalitate de dezvoltare a abilităților matematice și de îmbunătățire a capacității de gândire logică și analitică. Ele sunt utilizate în mod activ în educație, atât în școli, cât și în mediul online, pentru a stimula gândirea critică și rezolvarea de probleme la elevi.

În plus, puzzle-urile matematice au evoluat și în direcții inovatoare, încorporând concepte din alte domenii, cum ar fi informatica, criptografia sau chiar inteligența artificială. Această abordare a adus o nouă dimensiune în rezolvarea puzzle-urilor matematice, oferind provocări mai complexe și captivante.

## 2.1 Originea și evoluția Mathdoku

Originea și evoluția Mathdoku reprezintă o poveste interesantă despre dezvoltarea și popularizarea acestui puzzle matematic captivant. Mathdoku, cunoscut și sub denumirea de KenKen sau Calcudoku, a apărut pentru prima dată în Japonia la începutul anilor 2000 și a câștigat rapid popularitate în întreaga lume.

Conceptul de bază al Mathdoku se bazează pe Sudoku, un alt puzzle matematic extrem de popular. Mathdoku îmbină elemente de Sudoku cu principii matematice și operații aritmetice. A fost creat de matematicianul japonez Tetsuya Miyamoto, care a dorit să ofere o alternativă mai complexă și diversificată la puzzle-urile Sudoku. [1]

Primele variante de Mathdoku au fost publicate în presa japoneză sub numele de "Kashikoku naru Puzzle" (în traducere liberă "Puzzle-uri care te fac mai inteligent"). Acestea au captat atenția publicului prin abordarea inovatoare a rezolvării puzzle-urilor prin intermediul calculelor matematice și a gândirii logice.

Ulterior, Mathdoku a fost popularizat și redenumit sub numele de KenKen în Statele Unite, de către jurnalistul american și puzzlemaker-ul Will Shortz. El a întâlnit puzzle-ul în Japonia și a fost impresionat de potențialul său de a atrage atenția publicului internațional. Astfel, el a adus Mathdoku/KenKen în atenția publicului american prin intermediul publicației sale, New York Times. [2]

De atunci, Mathdoku a evoluat și s-a răspândit rapid în întreaga lume. A câștigat popularitate în rândul pasionaților de puzzle-uri matematice, fiind prezent în reviste specializate, cărți de puzzle-uri și în formă digitală, prin intermediul aplicațiilor și platformelor online.

Evoluția Mathdoku a inclus dezvoltarea de variante și niveluri de dificultate, care au adus o diversitate și provocări suplimentare în rezolvarea acestui puzzle. Astăzi, există o varietate de puzzle-uri Mathdoku disponibile, de la cele simple și ușor de rezolvat, până la cele extrem de complexe și provocatoare, destinate experților în matematică.

Originea și evoluția Mathdoku demonstrează că acest puzzle matematic a reușit să-și găsească un loc special în lumea puzzle-urilor, oferind o combinație unică de matematică, gândire logică și rezolvare de probleme. Popularitatea sa continuă să crească, iar Mathdoku rămâne unul dintre puzzle-urile matematice preferate și apreciate de către iubitorii de provocări intelectuale din întreaga lume.

## 2.2 Regulile și variantele Mathdoku

Regulile de bază ale Mathdoku sunt relativ simple, dar permit o varietate de abordări și niveluri de dificultate. Obiectivul este de a completa o grilă cu cifre, respectând anumite constrângeri matematice. Iată regulile fundamentale ale Mathdoku:

* Grila: Mathdoku utilizează o grilă pătrată împărțită în celule mai mici. Dimensiunea grilei poate varia, dar cel mai des întâlnită este o grilă de șiruri pătrate de 3x3, 4x4 sau 6x6.
* Cifre: Grila conține cifre de la 1 la N, unde N reprezintă dimensiunea laturii grilei. De exemplu, într-o grilă 4x4, veți avea cifrele de la 1 la 4.
* Regiuni: Grila este împărțită în regiuni mai mici, fiecare conținând un număr variabil de celule. Aceste regiuni pot fi pătrate sau rectangulare și sunt delimitate printr-o linie mai groasă.
* Constrângeri matematice: În fiecare regiune, trebuie să aplicați o anumită operație matematică (adunare, scădere, înmulțire sau împărțire) la cifrele din regiune pentru a obține un anumit rezultat. Acest rezultat este afișat în colțul stânga sus al regiunii.
* Celulele și restricții: Fiecare celulă din grilă trebuie să conțină o cifră unică și să respecte restricțiile impuse de operațiile matematice. Cifrele nu pot fi repetate în același rând sau coloană și trebuie să se potrivească cu rezultatele specificate în regiuni.

Variantele Mathdoku adaugă diferite elemente și provocări la regulile de bază. Acestea pot include:

* Mathdoku cu indici: Unele variante permit utilizarea indiciilor matematice în interiorul regiunilor pentru a indica operația matematică necesară sau relațiile între cifre.
* Mathdoku cu restricții aritmetice: Anumite variante pot adăuga restricții suplimentare, cum ar fi "cifrele adiacente să difere cu 1" sau "cifrele să fie în ordine crescătoare sau descrescătoare".
* Mathdoku cu cifre negative: Această variantă permite introducerea cifrelor negative în grilă și aplicarea operațiilor matematice corespunzătoare.
* Mathdoku cu joker: Unele variante includ jokeri, care pot fi înlocuiți cu orice cifră, respectând totuși regulile de bază.

## 2.3 Beneficiile rezolvării de puzzle-uri Mathdoku

Rezolvarea de puzzle-uri Mathdoku poate aduce o serie de beneficii și avantaje, atât din punct de vedere al dezvoltării intelectuale, cât și al divertismentului. Iată câteva dintre beneficiile asociate rezolvării de puzzle-uri Mathdoku:

* Dezvoltarea abilităților matematice: Mathdoku implică utilizarea principiilor matematice și a operațiilor aritmetice în rezolvarea puzzle-urilor. Acest lucru contribuie la îmbunătățirea abilităților de calcul mental, a rezolvării de probleme și a gândirii analitice.
* Stimularea gândirii logice: Rezolvarea puzzle-urilor Mathdoku necesită gândire logică și raționament deductiv. Jucătorii trebuie să analizeze situația și să identifice strategii adecvate pentru a completa grila în conformitate cu regulile și constrângerile specifice.
* Îmbunătățirea concentrării și a atenției: Rezolvarea unui puzzle Mathdoku necesită o atenție concentrată și focalizată pentru a identifica modele, a face conexiuni și a lua decizii corecte. Practicarea regulată a rezolvării de puzzle-uri Mathdoku poate contribui la dezvoltarea și îmbunătățirea capacității de concentrare și a atenției selective.
* Reducerea stresului și relaxarea: Rezolvarea puzzle-urilor Mathdoku poate fi un mijloc excelent de relaxare și distracție. Aceasta oferă o pauză de la rutina zilnică, permițând jucătorilor să se concentreze pe o activitate plăcută și captivantă, care poate induce o stare de relaxare și de destindere.
* Diversitate și adaptabilitate: Mathdoku vine în diferite niveluri de dificultate și variante, permițând jucătorilor să aleagă puzzle-uri adecvate abilităților și preferințelor lor. Această diversitate permite o adaptabilitate și o creștere progresivă a provocărilor, ceea ce încurajează jucătorii să-și dezvolte abilitățile și să își testeze limitele.
* Stimularea creierului și întreținerea funcțiilor cognitive: Rezolvarea de puzzle-uri matematice, cum ar fi Mathdoku, poate stimula activitatea cerebrală și poate contribui la întreținerea funcțiilor cognitive. Aceasta poate ajuta la menținerea agilității mentale și la prevenirea declinului cognitiv asociat cu îmbătrânirea.

Rezolvarea de puzzle-uri Mathdoku poate oferi astfel beneficii cognitive, emoționale și de divertisment. Indiferent de nivelul de experiență matematică, rezolvarea acestor puzzle-uri poate fi o activitate plăcută și captivantă, oferind oportunitatea de a exersa abilitățile matematice și de a îmbunătăți capacitățile cognitive. [3]

# Capitolul 3. Concepte, fundamente și tehonologii utilizate

În cadrul acestui capitol, vom explora conceptele esențiale ale Mathdoku și vom analiza fundamentele matematice care stau în spatele acestui joc captivant. Vom investiga, de asemenea, tehnologiile utilizate în dezvoltarea aplicației Mathdoku și vom examina rolul lor în crearea unei experiențe interactive și plăcute pentru utilizatori.

Prin explorarea conceptelor de bază, vom dezvălui esența Mathdoku și vom înțelege regulile și variantele acestui joc matematic. Vom explora, de asemenea, beneficiile rezolvării de puzzle-uri Mathdoku și impactul pozitiv pe care acestea îl pot avea asupra dezvoltării abilităților matematice și logice.

De asemenea, vom analiza fundamentele matematice care stau la baza Mathdoku, privind modul în care pot fi rezolvate și abordările matematice utilizate în acest proces. Vom explora algoritmii și strategiile de rezolvare, aducând în prim-plan aspectele matematice și logice implicate în rezolvarea eficientă a puzzle-urilor Mathdoku.

## 3.1 Fundamente matematice ale Mathdoku

Mathdoku se bazează pe fundamente matematice solide, iar rezolvarea acestor puzzle-uri implică aplicarea principiilor matematice și a unor strategii specifice. În această secțiune, vom explora fundamentele matematice ale Mathdoku și modul în care acestea sunt utilizate în rezolvarea puzzle-urilor. [4]

Operații matematice: Puzzle-urile Mathdoku implică utilizarea operațiilor matematice de bază, cum ar fi adunarea, scăderea, înmulțirea și împărțirea. Aceste operații sunt aplicate între cifrele dintr-o regiune și trebuie să conducă la rezultatul specificat în acea regiune.

Proprietăți matematice: Mathdoku se bazează pe diverse proprietăți matematice, cum ar fi asociativitatea, comutativitatea și distributivitatea, care pot fi utilizate pentru a găsi combinații corecte de cifre într-o regiune. De exemplu, înmulțirea este asociativă, ceea ce înseamnă că ordinea în care sunt efectuate operațiile de înmulțire nu contează.

Rezolvarea de ecuații și sisteme de ecuații: În Mathdoku, jucătorii trebuie să rezolve ecuații și sisteme de ecuații simple pentru a determina valorile cifrelor dintr-o regiune. Aceasta implică aplicarea principiilor de echivalență matematică și deducerea valorilor corecte.

Deducție logică: Rezolvarea puzzle-urilor Mathdoku implică, de asemenea, utilizarea deducției logice pentru a determina valorile corecte. Jucătorii pot utiliza restricțiile și informațiile furnizate de regiuni adiacente pentru a deduce cifrele potrivite.

Strategii și metode de rezolvare: Există o varietate de strategii și metode de rezolvare utilizate în Mathdoku, cum ar fi metoda "cifra singură", care implică identificarea cifrelor unice într-o regiune, sau metoda "excluderea valorilor", care implică eliminarea valorilor imposibile dintr-o regiune în funcție de constrângerile matematice.

Prin aplicarea fundamentelor matematice și a strategiilor de rezolvare adecvate, jucătorii pot aborda cu succes puzzle-urile Mathdoku și pot găsi soluții corecte. Rezolvarea acestor puzzle-uri nu doar îmbunătățește abilitățile matematice, ci și dezvoltă gândirea logică, deducția și abilitățile de rezolvare a problemelor.

## 3.2 Tehnologii utilizate în dezvoltarea aplicației Mathdoku

În dezvoltarea aplicației Mathdoku, unele dintre tehnologiile comune utilizate includ:

* JavaFX: JavaFX este o platformă de dezvoltare pentru crearea aplicațiilor desktop și mobile în limbajul de programare Java. Este frecvent utilizată pentru crearea interfețelor grafice de utilizator (GUI) în aplicația Mathdoku, oferind componente și funcționalități pentru afișarea și manipularea grafică a grilelor, celulelor și elementelor de interfață. [10]
* IntelliJ IDEA: IntelliJ IDEA este un mediu de dezvoltare integrat (IDE) utilizat pentru dezvoltarea aplicațiilor Java. Acesta oferă un set complet de instrumente pentru dezvoltatori, inclusiv funcționalități avansate de editare, depanare și construire a aplicațiilor JavaFX pentru acest proiect. [11]
* Java: Limbajul de programare Java este utilizat pentru a scrie codul sursă al aplicației Mathdoku. Java oferă o sintaxă clară și o platformă puternică, ceea ce face ca dezvoltarea aplicației Mathdoku să fie ușoară și flexibilă. [9]

Acestea sunt câteva tehnologii utilizate în dezvoltarea aplicației Mathdoku cu JavaFX în IntelliJ IDEA. Utilizarea acestor tehnologii permite dezvoltarea și crearea aplicației Mathdoku într-un mod interactiv și estetic, care oferă o experiență captivantă utilizatorilor.

## 3.3 Utilizarea Rapid7 în dezvoltarea aplicației Mathdoku

Rapid7 este o platformă puternică și versatilă de securitate și de analiză a vulnerabilităților, care poate fi utilizată în dezvoltarea aplicației Mathdoku pentru a asigura securitatea și fiabilitatea acesteia. Integrarea Rapid7 în procesul de dezvoltare aduce numeroase beneficii și instrumente utile pentru a identifica și remedia eventuale vulnerabilități sau probleme de securitate. În acest subcapitol, vom explora modul în care utilizarea Rapid7 poate contribui la dezvoltarea și protecția aplicației Mathdoku.

Testare de securitate: Rapid7 oferă un set de instrumente și soluții de testare a securității, cum ar fi Metasploit și Nexpose, care pot fi utilizate pentru a identifica eventuale vulnerabilități sau expuneri la atacuri în aplicații Mathdoku. Aceste instrumente permit testarea și evaluarea securității aplicației, identificarea punctelor slabe și luarea de măsuri pentru remedierea lor.

Monitorizare a vulnerabilităților: Utilizând soluții Rapid7, precum InsightVM, dezvoltatorii pot monitoriza și gestiona vulnerabilitățile aplicației Mathdoku într-un mod centralizat și eficient. Aceasta permite identificarea și remedierea rapidă a problemelor de securitate înainte ca acestea să fie exploatate de potențiali atacatori.

Securitate în ciclul de dezvoltare: Integrarea Rapid7 în procesul de dezvoltare al aplicației Mathdoku permite implementarea unor practici solide de securitate în întregul ciclu de viață al aplicației. Acest lucru include testarea continuă a securității, identificarea și remedierea promptă a vulnerabilităților, precum și aplicarea de politici și standarde de securitate adecvate.

Analiza de securitate și rapoarte: Rapid7 oferă capacități avansate de analiză și raportare în domeniul securității. Prin utilizarea acestor instrumente, dezvoltatorii pot genera rapoarte detaliate despre vulnerabilitățile identificate, nivelurile de risc și măsurile de remediere luate. Aceste rapoarte oferă o imagine cuprinzătoare a stării de securitate a aplicației Mathdoku și facilitează luarea de decizii informate pentru îmbunătățirea securității.

Utilizarea Rapid7 în dezvoltarea aplicației Mathdoku aduce beneficii semnificative în ceea ce privește securitatea și protecția datelor. Aceasta permite dezvoltatorilor să identifice și să remedieze rapid vulnerabilitățile, să implementeze măsuri solide de securitate și să ofere o experiență sigură utilizatorilor. Integrarea Rapid7 în procesul de dezvoltare este un pas important pentru a asigura că aplicațiile Mathdoku sunt robuste, fiabile și protejate împotriva potențialelor amenințări de securitate.

## 3.4 Utilizarea ALM în dezvoltarea aplicației Mathdoku

În cadrul dezvoltării aplicației Mathdoku, ALM joacă un rol important în gestionarea procesului de dezvoltare. Prin utilizarea ALM, se pot gestiona cerințele aplicației, se poate realiza gestionarea configurației și versionarea codului sursă, se pot defini și urmăriți sarcinile și defectele, se poate realiza testarea și verificarea, și se poate realiza implementarea și livrarea aplicației.

Prin aplicarea unui proces eficient de ALM, se poate asigura o dezvoltare mai rapidă și mai organizată a aplicației Mathdoku, se pot minimiza riscurile și erorile și se poate asigura o experiență de utilizare mai bună pentru utilizatori.

În concluzie, utilizarea ALM în dezvoltarea aplicației Mathdoku a fost esențială pentru gestionarea și coordonarea eficientă a activităților de dezvoltare și pentru asigurarea calității și fiabilității aplicației pe parcursul întregului ciclu de viață al acesteia.

# Capitolul 4. Dezvoltarea aplicației Mathdoku

Această lucrare are ca obiectiv proiectarea și implementarea unui joc numit “Mathdoku”. Dezvoltarea s-a realizat pe baza tehnologiilor prezentate în capitolele anterioare.

Jocul este în totalitate dezvoltat în limbajul de programare JavaFX, care mi-a oferit beneficiul de a utiliza un set puternic de instrumente pentru construirea interfețelor utilizator vizual atrăgătoare și interactive care au făcut aplicația cât mai plăcut de utilizat.

Acest capitol se concentrează pe procesul de dezvoltare al aplicației Mathdoku, un joc de puzzle matematic captivant. Vom explora etapele cheie ale dezvoltării, de la analiza și proiectarea aplicației, până la implementarea și optimizarea acesteia. Scopul acestui capitol este de a oferi o imagine detaliată asupra modului în care aplicația Mathdoku a fost dezvoltată, evidențiind aspectele relevante ale fiecărei etape.

## 4.1 Analiza aplicației Mathdoku

În această etapă, am efectuat o analiză detaliată a aplicației Mathdoku pentru a identifica cerințele funcționale ale acesteia. Am investigat caracteristicile-cheie ale jocului și am examinat nevoile și preferințele potențialilor utilizatori.

Pentru a înțelege mai bine scopul și funcționalitățile aplicației Mathdoku, am analizat puzzle-urile matematice și regulile specifice care le guvernează. Am studiat variantele și nivelurile de dificultate existente în jocul Mathdoku și am explorat cum pot fi implementate în aplicație.

În cadrul analizei, am identificat următoarele cerințe funcționale ale aplicației Mathdoku:

* Interfața utilizatorului intuitivă: Aplicația oferă o interfață utilizator prietenoasă și intuitivă care să permită utilizatorului să interacționeze ușor cu puzzle-urile și să introducă răspunsurile. Răspunsurile se pot introduce de la tastatura sau folosind butoanele din partea dreaptă a aplicației.
* Sistem de verificare a răspunsurilor: Aplicația verifică răspunsurile introduse de utilizator prin apăsarea butonului “Show mistakes”.
* Schimbarea dificultății puzzle-ului: Aplicația oferă o multitudine de dificultați, utilizatorul poate alege să rezolve puzzle-uri de dimensiune 2x2 pana la dimensiune de 8x8. Aceste puzzle-uri pot fi alese prin apăsarea butonului “Load game from file”.
* Sistem de sugestii: Aplicația oferă sugestii in rezolvarea puzzle-urilor prin butonul “hint”.
* Sistem de schimbare a fontului: Aplicația oferă posibilitatea de a schimba mărimea fontului pentru a aduce o experientă cat mai plăcută in utilizarea acesteia.
* Sistem de adăugare puzzle: Aplicația oferă posibilitatea de a crea noi puzzle-uri de către utilizator prin introducerea acestora de la tastatura prin butonul “Load game from text input”.

## 4.2 Proiectarea aplicației Mathdoku

Proiectarea presupune realizarea unui plan sau a unei schițe în funcție de cerințele aplicației, urmând ca la partea de implementare să fie folosite toate conceptele, fundamentele și ideile de la proiectare și puse în aplicare.

Aplicația a fost gândită si proiectată într-un mod care o face ușor de folosit și pentru a satisface orice fel de utilizator prin implementarea mai multor nivele de dificultate.

Design conceptual:

* Identificarea cerințelor: Stabilirea funcționalităților principale ale aplicației Mathdoku, cum ar fi rezolvarea puzzle-urilor, interacțiunea cu utilizatorul, gestionarea nivelurilor de dificultate etc.
* Definirea obiectivelor: Stabilirea obiectivelor pe care aplicația Mathdoku trebuie să le îndeplinească, cum ar fi oferirea unei experiențe de joc captivante, rezolvarea eficientă a puzzle-urilor, asigurarea unei interfețe intuitive pentru utilizatori etc.

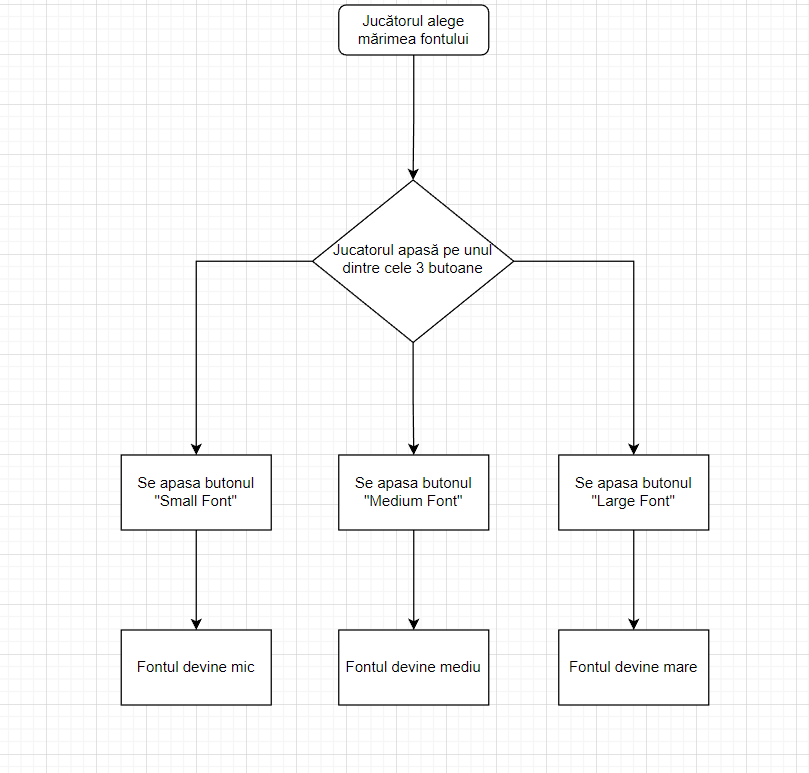


Figure 1

* Butonul "Small Font":

Scop: Permite utilizatorilor să selecteze o dimensiune mai mică pentru fontul utilizat în interfață.

Funcționalitate: La apăsarea butonului "Small Font", dimensiunea fontului în interfață va fi redusă. Acest lucru poate fi util pentru utilizatorii care doresc o dimensiune mai mică a textului sau care doresc să optimizeze spațiul de afișare.

* Butonul "Medium Font":

Scop: Permite utilizatorilor să selecteze o dimensiune medie pentru fontul utilizat în interfață.

Funcționalitate: La apăsarea butonului "Medium Font", dimensiunea fontului în interfață va fi setată la o dimensiune medie. Această opțiune oferă un echilibru între dimensiunea mică și cea mare a fontului, potrivindu-se preferințelor majorității utilizatorilor.

* Butonul "Large Font":

Scop: Permite utilizatorilor să selecteze o dimensiune mai mare pentru fontul utilizat în interfață.

Funcționalitate: La apăsarea butonului "Large Font", dimensiunea fontului în interfață va fi mărită. Acest lucru poate fi benefic pentru utilizatorii cu deficiențe de vedere sau care preferă dimensiuni mai mari pentru o citire ușoară.

Prin includerea acestor butoane în interfața aplicației, utilizatorii pot personaliza dimensiunea fontului în funcție de preferințele lor individuale. Aceasta asigură o experiență de utilizare mai plăcută și mai accesibilă, deoarece utilizatorii pot ajusta fontul în funcție de nevoile lor de vizualizare. Fiecare buton oferă o opțiune distinctă și flexibilitate în alegerea dimensiunii fontului, permițând utilizatorilor să se adapteze confortabil la aplicație.

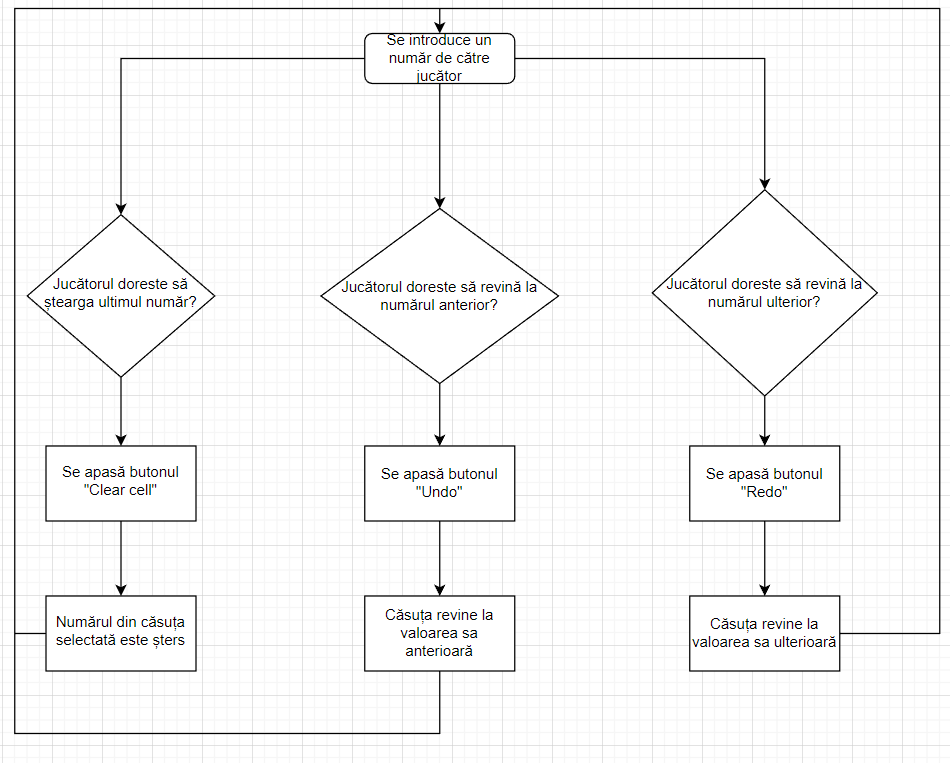


Figure 2

* Butonul "Clear Cell":

Scop: Permite utilizatorului să șteargă conținutul unei celule selectate în tabla de joc.

Funcționalitate: La apăsarea butonului "Clear Cell", conținutul celulei selectate va fi șters, permițând utilizatorului să reintroducă o nouă valoare în acea celulă.

* Butonul "Undo":

Scop: Oferă utilizatorului posibilitatea de a anula ultima acțiune efectuată în rezolvarea puzzle-ului Mathdoku.

Funcționalitate: La apăsarea butonului "Undo", ultima acțiune a utilizatorului va fi anulată, restaurând starea anterioară a puzzle-ului. Aceasta poate include ștergerea unei valori introduse, modificarea unei valori sau alte acțiuni relevante.

* Butonul "Redo":

Scop: Permite utilizatorului să refacă o acțiune care a fost anulată utilizând butonul "Undo".

Funcționalitate: La apăsarea butonului "Redo", acțiunea anterioară anulată prin utilizarea butonului "Undo" va fi refăcută, restabilind astfel starea puzzle-ului la o versiune actualizată.

Aceste butoane sunt esențiale pentru a oferi utilizatorilor flexibilitate și control în rezolvarea puzzle-urilor Mathdoku. Prin includerea acestor butoane în interfața aplicației, utilizatorii pot corecta erori, reveni la stări anterioare și continua rezolvarea puzzle-ului cu ușurință și încredere.

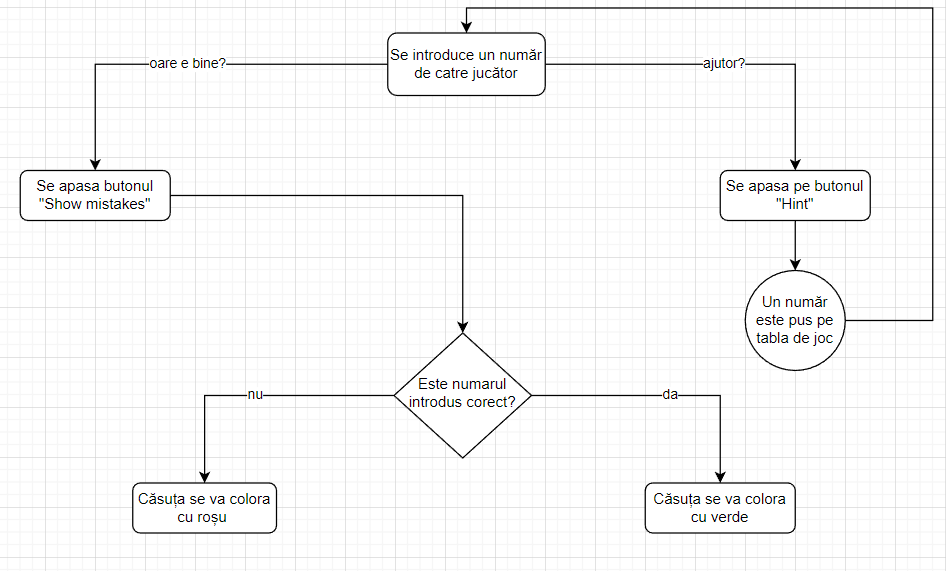


Figure 3

* Butonul "Hint":

Scop: Furnizează utilizatorului un indiciu pentru a-l ghida în rezolvarea puzzle-ului Mathdoku.

Funcționalitate: La apăsarea butonului "Hint", aplicația va furniza utilizatorului un indiciu în rezolvarea puzzle-ului.

* Butonul "Show Mistakes":

Scop: Ajută utilizatorul să identifice și să evidențieze greșelile făcute în rezolvarea puzzle-ului Mathdoku.

Funcționalitate: La apăsarea butonului "Show Mistakes", aplicația va analiza puzzle-ul rezolvat de către utilizator și va evidenția celulele care conțin valori greșite. Celulele cu greșeli sunt evidențiate cu roșu pentru a atrage atenția utilizatorului și a-l ajuta să identifice și să corecteze erorile. Celulele completate corect sunt evidențiate cu verde.

Aceste butoane suplimentare oferă utilizatorilor suport adițional în rezolvarea puzzle-urilor Mathdoku. Butonul "Hint" îi poate ghida pe utilizatori atunci când se confruntă cu dificultăți în rezolvare, în timp ce butonul "Show Mistakes" le permite să identifice și să corecteze erorile pentru a obține o soluție corectă și completă. Prin includerea acestor butoane în interfața aplicației, utilizatorii beneficiază de un instrumentar mai amplu pentru a rezolva puzzle-urile Mathdoku cu succes.

## 4.3 Implementarea aplicației Mathdoku

Implementarea aplicației Mathdoku a implicat codificarea funcționalităților și a interfeței utilizator. Am utilizat limbajul de programare Java și framework-ul JavaFX pentru a dezvolta aplicația. Am implementat logica de rezolvare a puzzle-urilor Mathdoku, gestionarea interacțiunii utilizatorului și integrarea funcționalităților suplimentare, precum niveluri de dificultate.

Interfața jocului este creată prin metoda "start". Metoda "start" este o metodă cheie în dezvoltarea unei aplicații JavaFX și este apelată automat de către platformă atunci când aplicația este pornită. În această metodă avem următoarele implementări:

* Se creează un VBox (dimensionVbox) care va conține butoanele pentru selectarea dimensiunii. Se setează alinierea verticală, distanța dintre componente și se adaugă o etichetă și trei butoane pentru selectarea dimensiunii fontului.

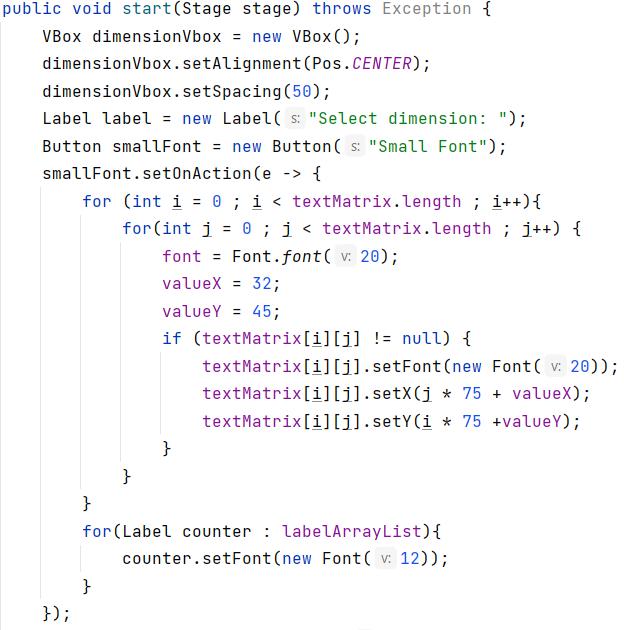


Figure 4

* Se creează un HBox (optionBox) care va conține butoanele pentru opțiuni (undo, redo, clear, show mistakes). Se setează alinierea orizontală, distanța dintre componente și se adaugă butoanele corespunzătoare.

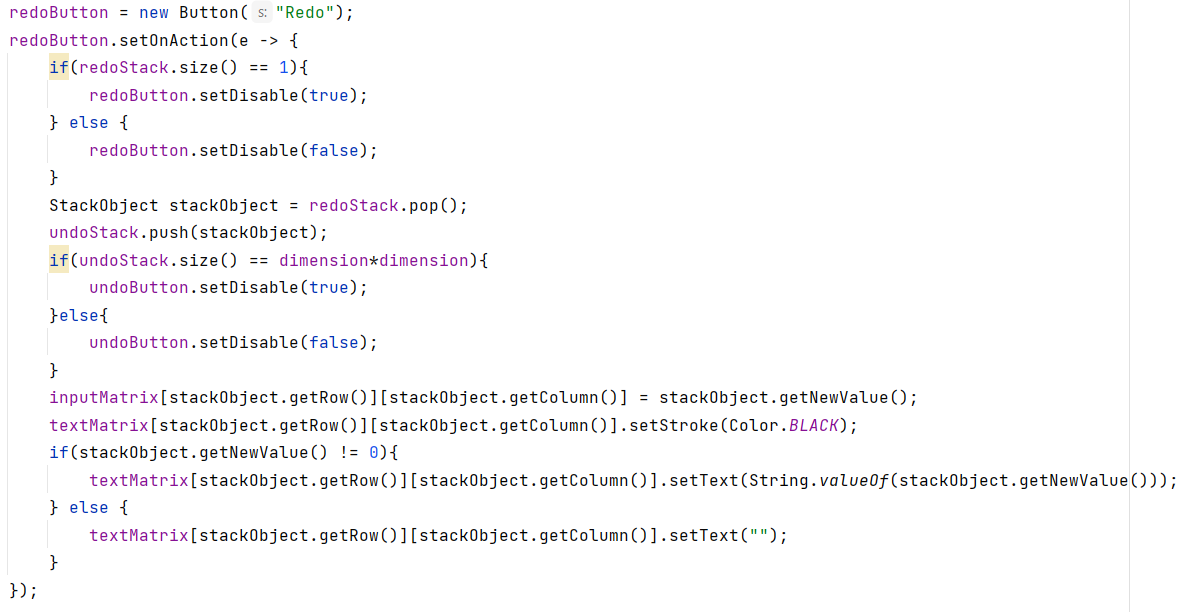


Figure 5

* Se creează un HBox (loadBox) care va conține butoanele pentru încărcarea jocului. Se setează alinierea orizontală, distanța dintre componente și se adaugă două butoane pentru încărcarea jocului din fișier sau dintr-un text introdus manual, precum și butoanele pentru rezolvarea jocului și obținerea de sugestii.

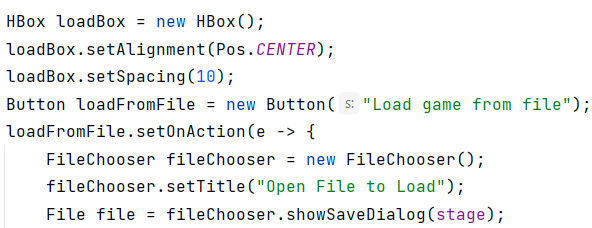


Figure 6

* Se creează un VBox (numberVbox) care va conține butoanele pentru selectarea numărului de introdus. Se setează alinierea verticală, distanța dintre componente și se adaugă o etichetă și butoanele de la 1 la 8, precum și un buton pentru ștergerea unei celule.

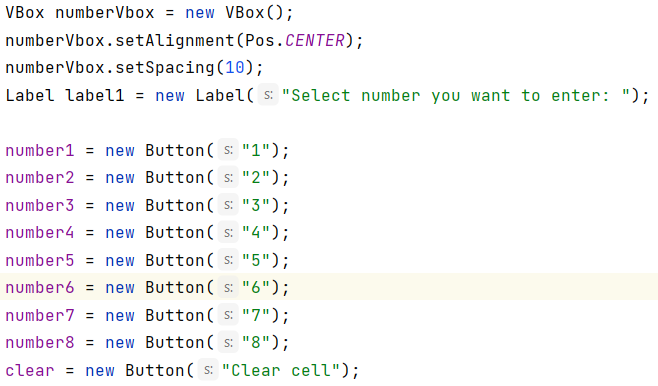


Figure 7

* Se setează dimensiunea tablei de joc (variabila "dimension" este inițializată cu valoarea 6).



Figure 8

* Se creează obiecte de tip Rectangle și se adaugă într-un ArrayList numit "recs". Aceste dreptunghiuri vor reprezenta celulele tablei de joc.



Figure 9

* Se apelează metodele "Cages()" și "createGrid()" pentru a crea grila de joc și pentru a afișa numerele de identificare ale zonelor.
* Se creează un obiect de tip FlowPane numit "wrap" și se adaugă grila de joc în el. FlowPane este utilizat pentru a aranja componentele într-un mod flexibil.

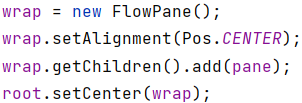


Figure 10

* Se setează acțiunile butoanelor și a tastelor pentru a gestiona interacțiunea utilizatorului cu aplicația.
* Se construiește scena și se afișează fereastra principală a aplicației.

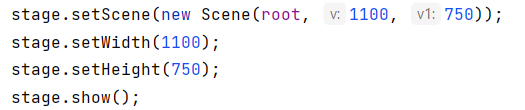


Figure 11

În continuare avem diverse funcții care au scopuri precum:

* Funcția createRectangles(int n) este responsabilă pentru crearea matricei de dreptunghiuri care reprezintă grila de joc. Se creează un număr de n x n dreptunghiuri și se adaugă la o listă rectangles. Fiecare dreptunghi are o acțiune de clic atașată, care schimbă culoarea dreptunghiului selectat și păstrează informații despre rândul și coloana selectate.



Figure 12

* Funcția createGrid(ArrayList<Shape> elem) primește o listă de forme și le adaugă la un panou, returnând panoul rezultat.

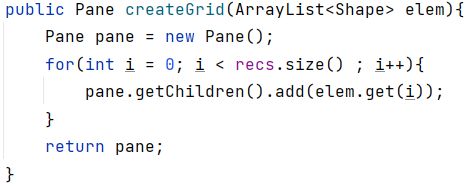


Figure 13

* Funcția createFileCages() este responsabilă pentru citirea și crearea cuștilor (cage-urilor) dintr-un fișier. Fiecare cușcă conține un rezultat și o listă de celule corespunzătoare.

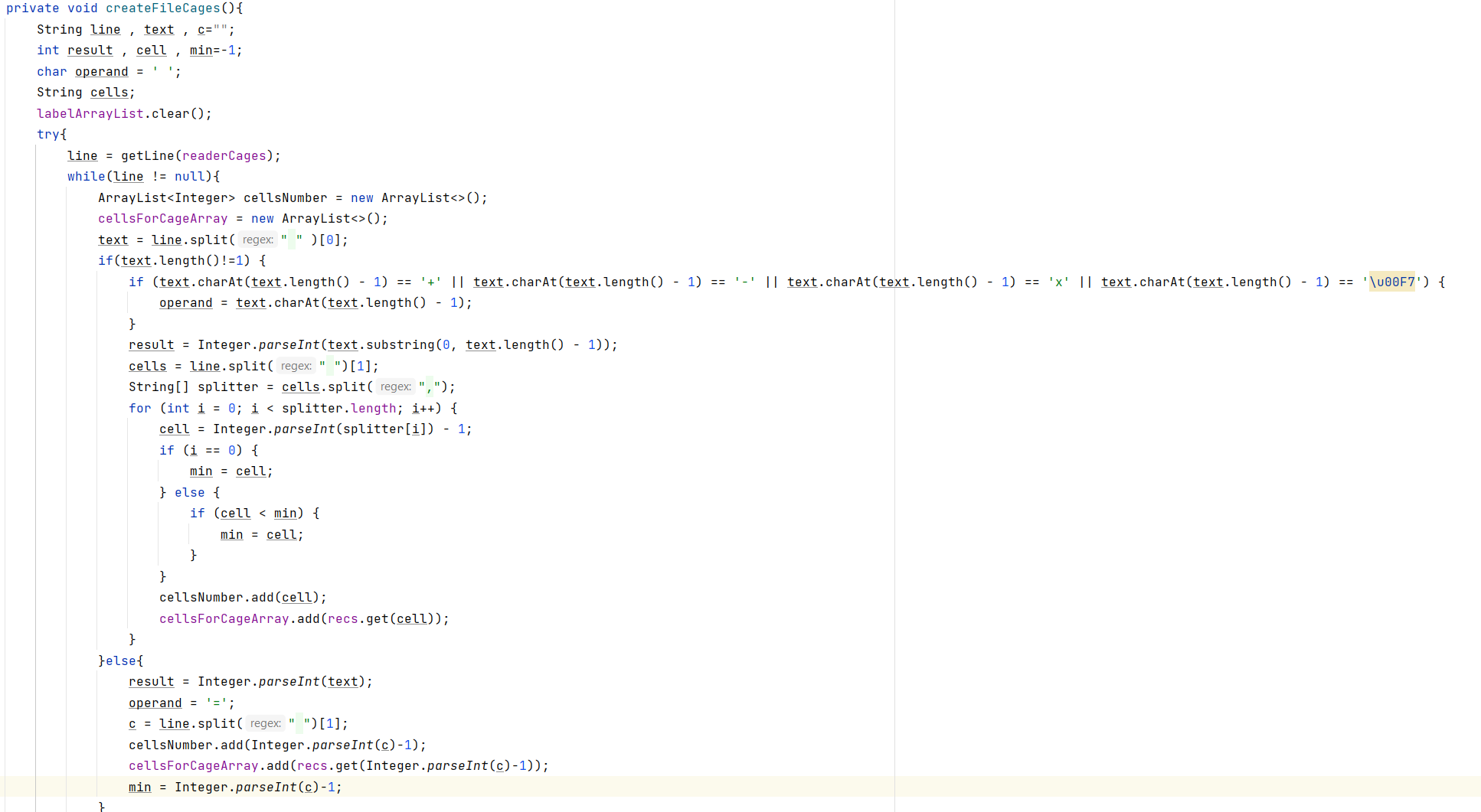


Figure 14

* Funcția createInputCages(String input) primește un șir de intrare și creează cuștile pe baza acestuia. Funcționalitatea este similară cu createFileCages(), dar datele sunt citite dintr-un șir de caractere în loc de fișier.

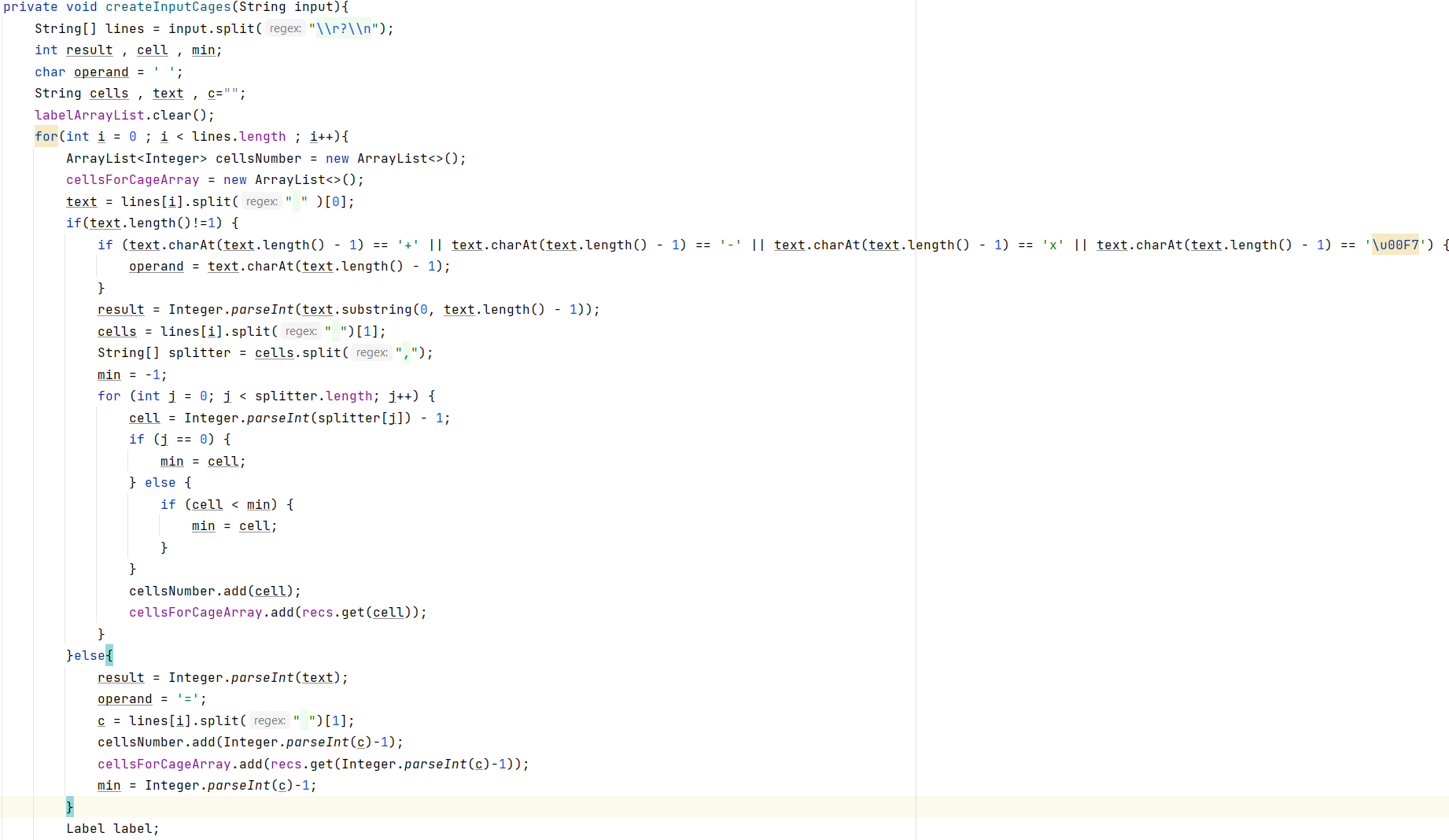


Figure 15

* Funcția Cages() este folosită pentru a crea cuștile direct în cod, fără a le citi dintr-un fișier sau șir de caractere.



Figure 16

Metodele prezente in aplicație sunt următoarele:

* setFileDimension(): Această metodă citește un fișier linie cu linie, procesează datele și setează variabila "dimension" pe baza valorii maxime găsite în fișier. Se așteaptă ca formatul fișierului să conțină linii cu două părți separate de spațiu: un număr urmat de un spațiu și o listă de numere separate prin virgulă.

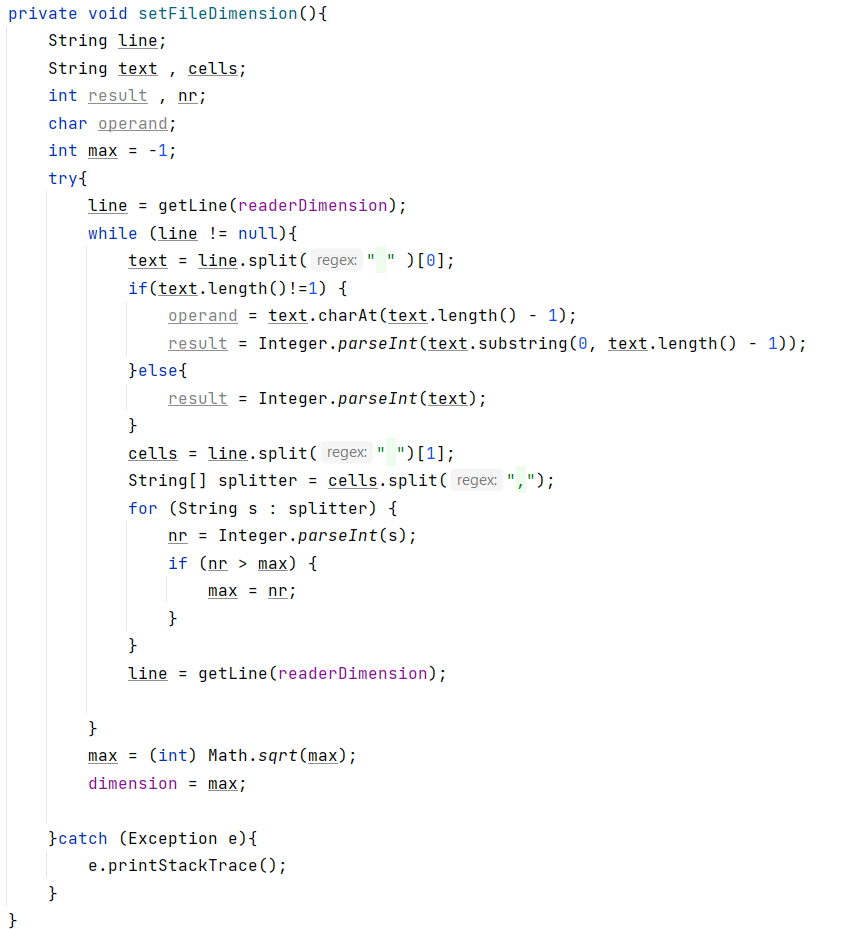


Figure 17

* setInputDimension(String input): Această metodă primește un input sub formă de șir de caractere, îl procesează similar cu metoda setFileDimension() și setează variabila "dimension" pe baza valorii maxime găsite în input. Se așteaptă ca formatul inputului să conțină linii cu două părți separate de spațiu: un număr urmat de un spațiu și o listă de numere separate prin virgulă.

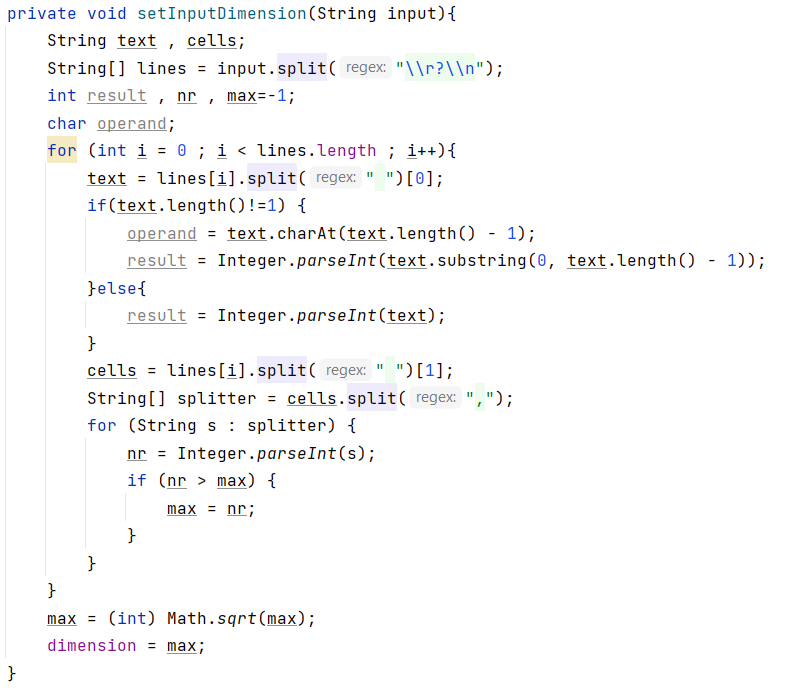


Figure 18

* createLabels(Shape shape, String text): Această metodă creează un obiect JavaFX Label cu textul dat și îl adaugă într-o listă (labelArrayList) și într-un panou (pane) pentru afișare.

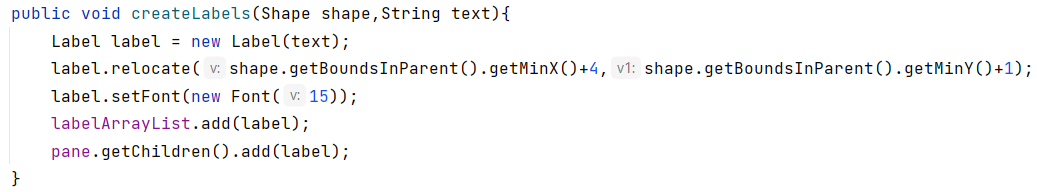


Figure 19

* displayLabels(): Această metodă apelează de mai multe ori metoda createLabels() pentru a crea și afișa etichete specifice în cage-uri.



Figure 20

* handleKeyPressEvent(): Această metodă gestionează evenimentele de apăsare a tastelor pentru aplicația JavaFX. Ascultă apăsările de taste pentru cifre (cu excepția cifrei 0) și tasta de backspace. Când este apăsată o tastă de cifră, se actualizează celula selectată în matrice cu cifra corespunzătoare. La apăsarea tastei backspace, se șterge conținutul celulei selectate în matrice.

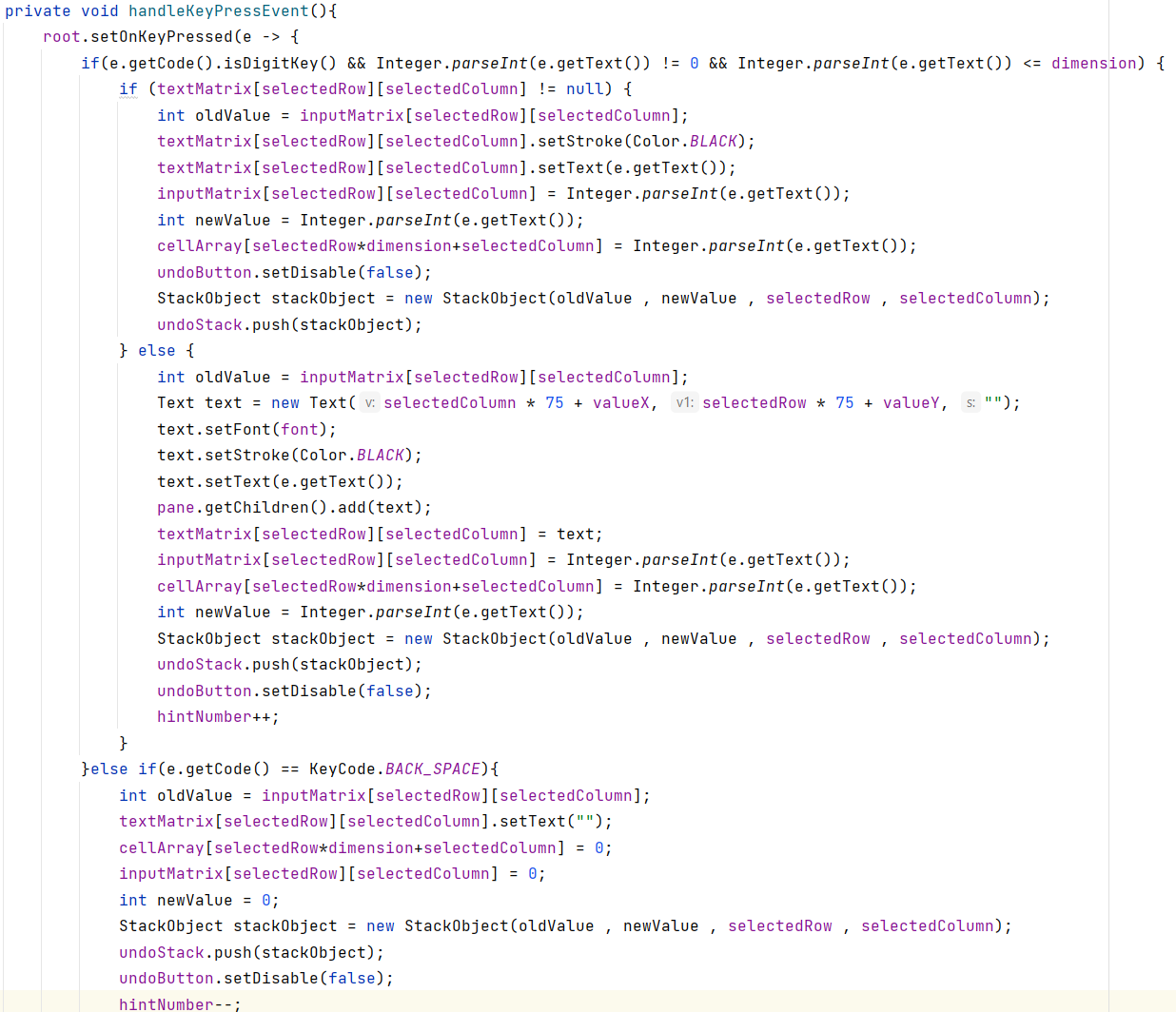


Figure 21

* setInputButtonsAction(): Această metodă configurează gestionarii de acțiuni pentru butoanele (number1, number2, number3, etc.). Când un buton este apăsat, se actualizează celula selectată în matrice cu cifra corespunzătoare.

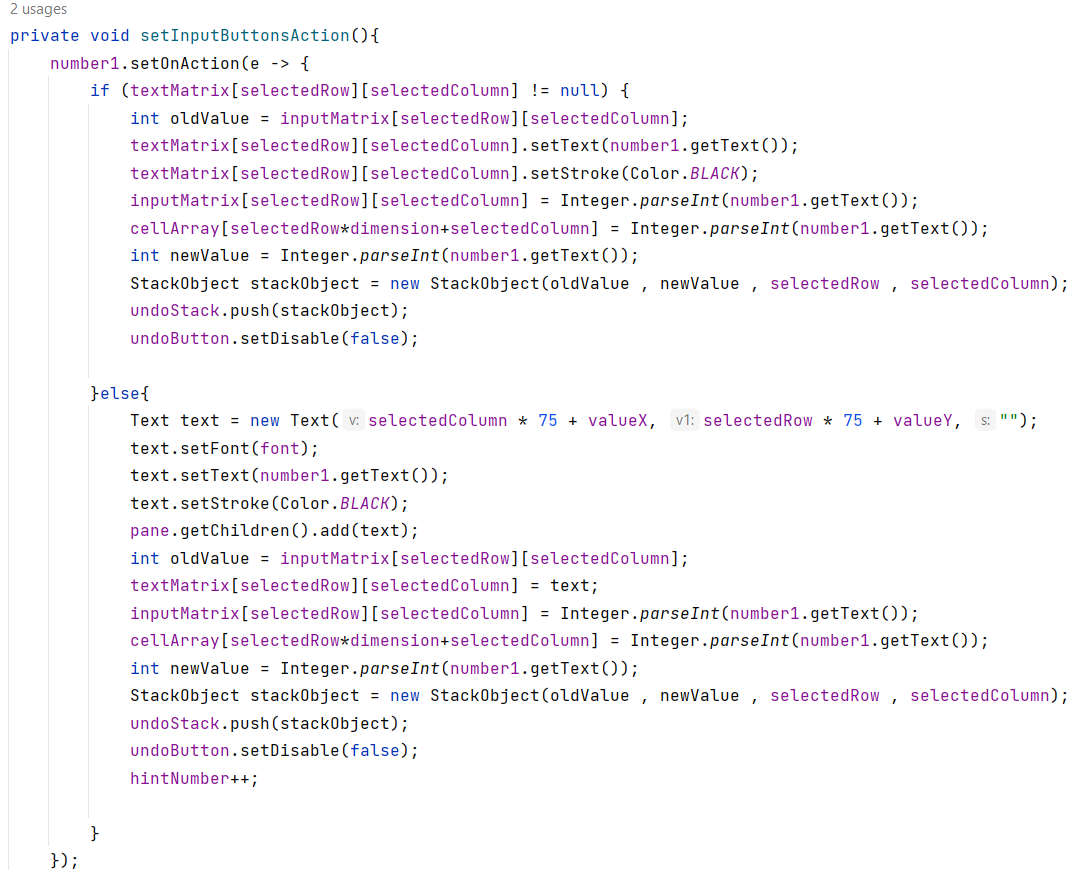


Figure 22

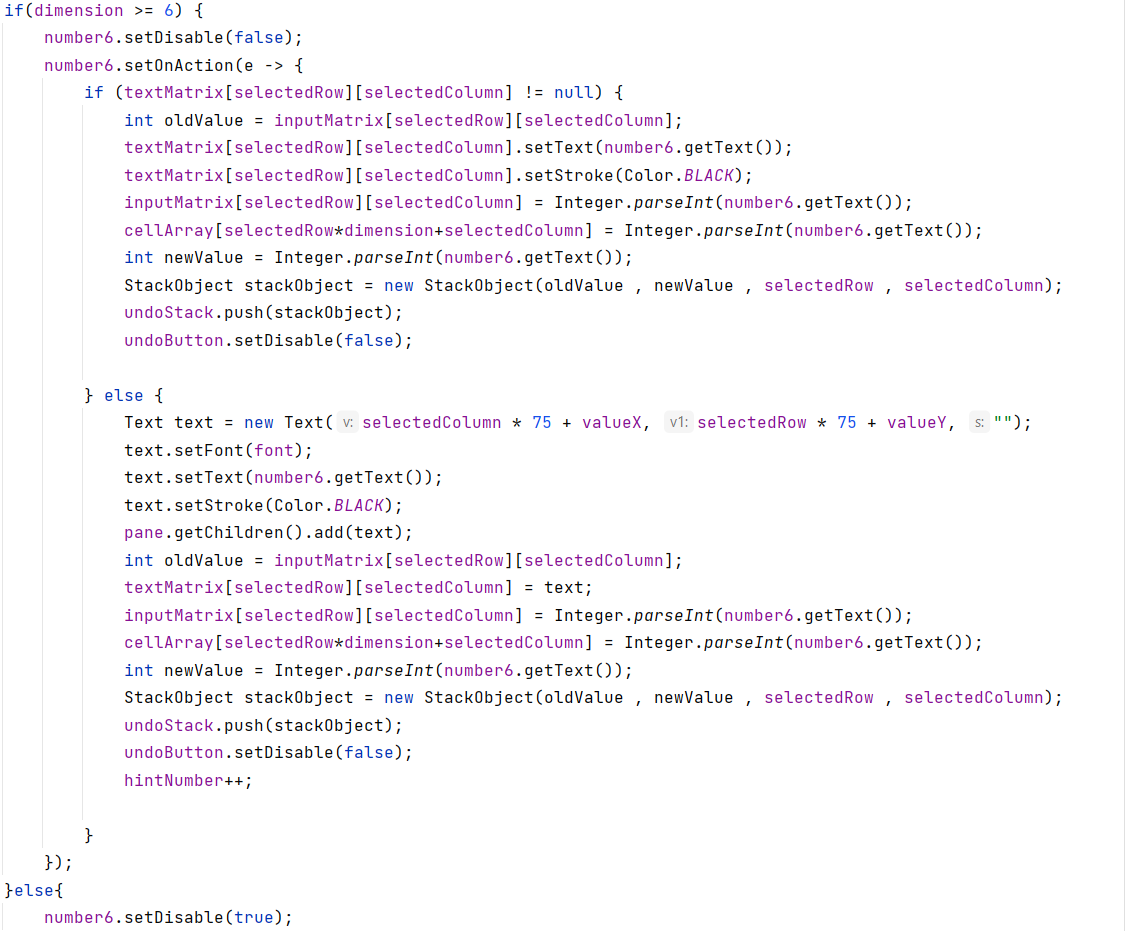


Figure 23

În cazul in care un joc de mărime 6x6 este deschis doar primele 6 butoane vor putea fi apăsate.



Figure 24

* Metoda "Mistakes()" verifică și evidențiază greșelile din puzzle, colorând dreptunghiurile corespunzătoare în roșu și afișând soluția corectă prin colorarea dreptunghiurilor în verde.



Figure 25

* getLine(BufferedReader reader): Această metodă citește o linie dintr-un obiect BufferedReader și o returnează ca un șir de caractere.

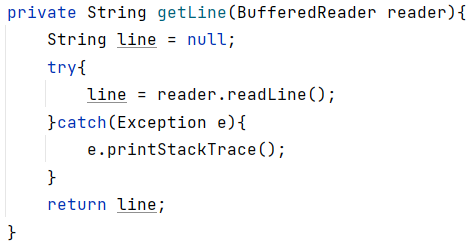


Figure 26

* checkCorrectFile(BufferedReader file): Această metodă verifică dacă fișierul de intrare conține date valide pentru rezolvarea puzzle-ului KenKen. Parcurge fiecare linie din fișier și aplică următoarele verificări:

1. Extrage celulele pentru fiecare cușcă din linie.
2. Verifică dacă celulele sunt valide în ceea ce privește poziționarea și frecvența.
3. Returnează true dacă fișierul este valid sau false în caz contrar.

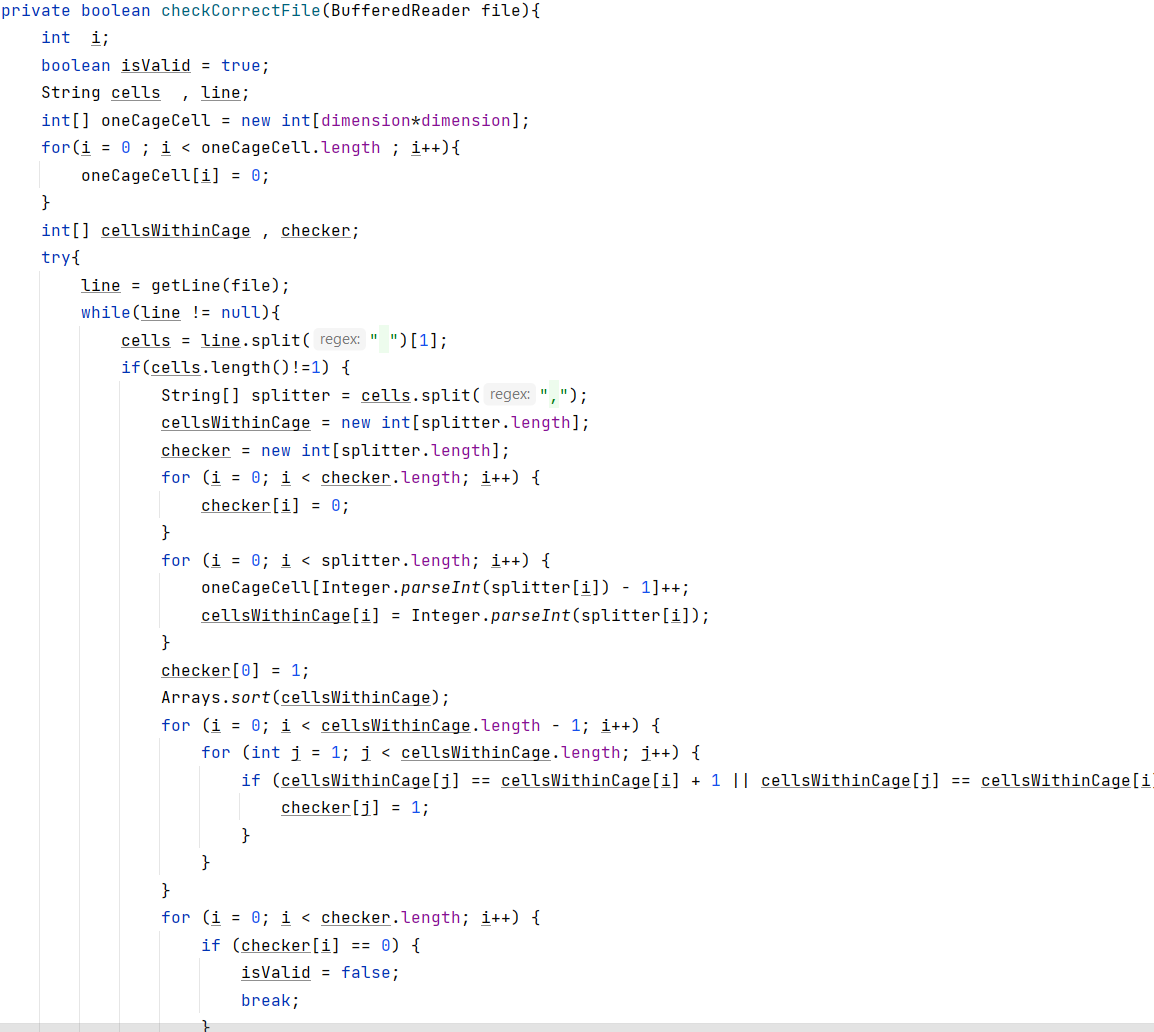


Figure 27

* checkCorrectInput(String input): Această metodă verifică dacă intrarea utilizatorului este validă pentru rezolvarea puzzle-ului KenKen. Parcurge fiecare linie de intrare și aplică aceleași verificări ca și în checkCorrectFile().

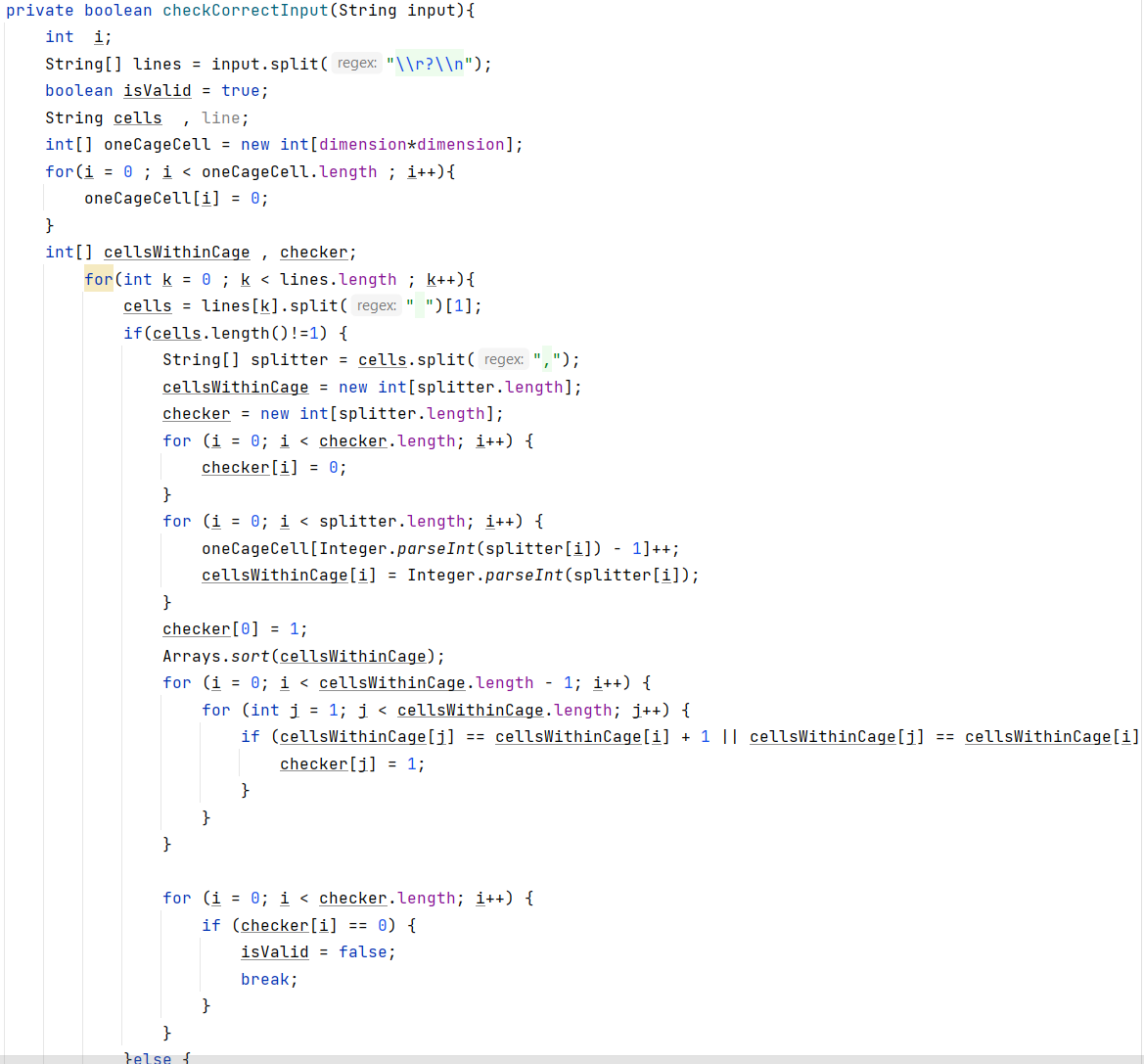


Figure 28

* createCageBoard(): Această metodă construiește o matrice cageBoard în care fiecare celulă este etichetată cu numărul cuștii corespunzătoare. Folosește lista allCages și valorile celulelor pentru a identifica cuștile din matricea de intrare.

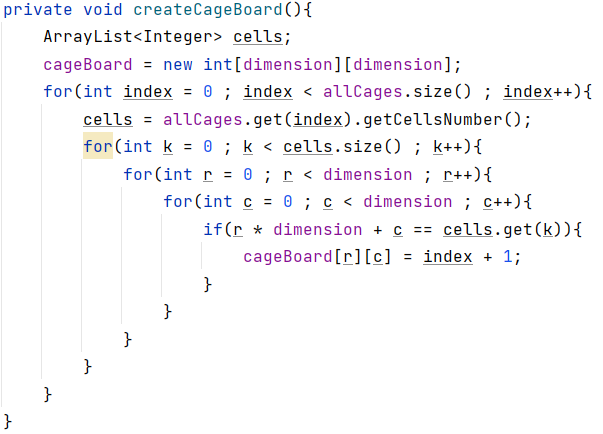


Figure 29

Cele cinci clase folosite în aplicatia Mathdoku au rolul de a:

1. **MathdokuGUI** este descris mai sus, aceasta este “main-ul” aplicației.
2. **Cage** este clasa utilizată pentru a reprezenta și manipula cuștile din puzzle-ul KenKen.

* private Shape cage: Variabila care stochează forma cuștii.
* private int target: Variabila care stochează rezultatul țintă pentru cușcă.
* private char operand: Variabila care stochează operatorul matematic utilizat în cușcă.
* private ArrayList<Integer> cellsNumber: Lista care stochează numerele celulelor din cușcă.
* private ArrayList<Shape> cageCells: Lista care stochează formele geometrică ale celulelor din cușcă.
* private ArrayList<Cage> allCages: Lista care stochează toate cuștile create.
* private ArrayList<Location> squares: Lista care stochează locațiile celulelor din cușcă.
* public Cage(int result, char operand, ArrayList<Shape> array, ArrayList<Integer> cellsNumber): Constructorul clasei "Cage" care primește rezultatul țintă, operatorul matematic, lista de forme geometrică ale celulelor și lista de numere ale celulelor și le atribuie variabilelor corespunzătoare.
* public Shape createCage(): Metoda care creează forma geometrică a cuștii pe baza formelor geometrice ale celulelor. Verifică dacă cușca are mai mult de o celulă și, în funcție de acest lucru, creează cuști complexe sau simple utilizând metoda Shape.union(). De asemenea, stabilește culorile și proprietățile vizuale ale cuștii și adaugă cuștile create în lista allCages. Returnează forma geometrică a cuștii.
* public char getOperand(): Metoda care returnează operatorul matematic al cuștii.
* public int getTarget(): Metoda care returnează rezultatul țintă al cuștii.
* public ArrayList<Integer> getCellsNumber(): Metoda care returnează lista de numere ale celulelor din cușcă.
* public void addSquares(Location location): Metoda care adaugă o locație de celulă în lista squares.



Figure 30

1. **Location** este o clasa simplă care este utilizată pentru a reprezenta și accesa o locație într-o grilă bidimensională, precum o coordonată într-un sistem de coordonate cartezian.

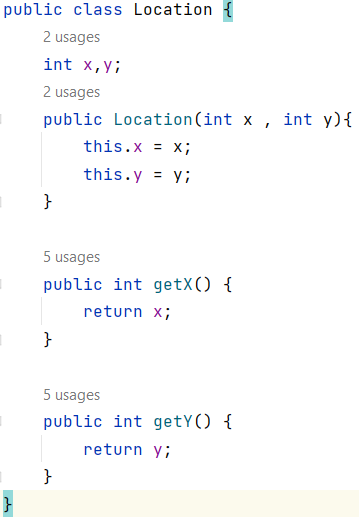


Figure 31

1. **Solver** este clasa care implementează algoritmul de rezolvare a puzzle-ului KenKen prin parcurgerea recursivă a tuturor posibilităților și verificarea restricțiilor impuse de regulile jocului.

* private int N, NCage: Variabilele care stochează dimensiunea tablei de joc (N) și numărul de custi (NCage).
* private Cage[] cages: Un vector de obiecte de tip "Cage" care reprezintă cuștile din puzzle.
* private int[][] board: Matricea care reprezintă tabla de joc și stochează valorile numerice.
* private int[][] cageBoard: Matricea care asociază fiecărei celule de pe tabla de joc cuști corespunzătoare.
* public Solver(int boardSize, Cage[] cages, int[][] cageBoard): Constructorul clasei "Solver" care primește dimensiunea tablei de joc, vectorul de cuști și matricea de asociere a cuștilor la celule.
* public void initialize(): Metoda care inițializează tabla de joc și adaugă celulele corespunzătoare fiecărei custi.
* public boolean solve(): Metoda recursivă care rezolvă puzzle-ul KenKen. Aceasta completează valorile în celulele goale ale tablei de joc și verifică dacă soluția este validă.
* public boolean isValid(int row, int col, int num): Metoda care verifică dacă o anumită valoare este validă pentru a fi plasată într-o anumită celulă.
* public boolean checkRow(int row, int num): Metoda care verifică dacă o anumită valoare există deja într-un rând.
* public boolean checkColumn(int col, int num): Metoda care verifică dacă o anumită valoare există deja într-o coloană.
* public boolean isFull(Cage cage): Metoda care verifică dacă o cușcă este completă, adică toate celulele sale sunt completate cu valori.
* public boolean checkCage(Cage cage): Metoda care verifică dacă o cușcă are rezultatul corect în funcție de valorile din celulele sale și operatorul specificat.
* public Location findEmpty(): Metoda care găsește o celulă goală în tabla de joc.
* public int[][] getBoard(): Metoda care returnează tabla de joc rezolvată.

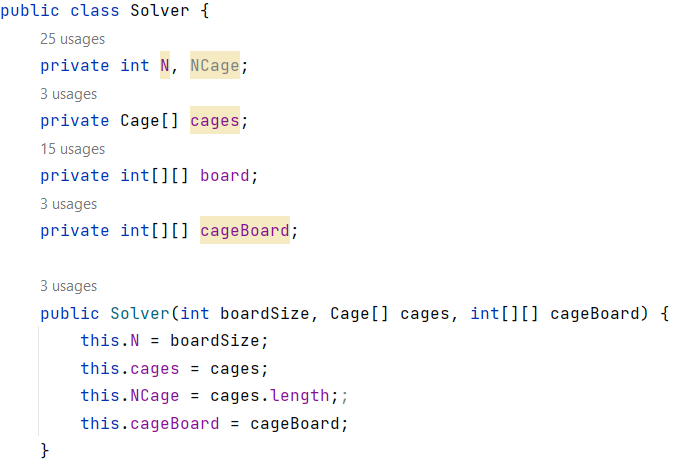


Figure 32

1. **StackObject** este clasa utilizată în implementarea unei stive pentru a gestiona și reveni la modificările efectuate în timpul rezolvării puzzle-ului KenKen. O instanță a clasei este creată atunci când o celulă este modificată, iar informațiile despre modificare sunt stocate în obiectul "StackObject". Aceasta permite utilizatorului să revină la valorile anterioare ale celulelor în cazul în care se ajunge într-o situație în care soluția nu este validă și este necesară revenirea la o stare anterioară a tablei de joc.

* int oldValue, newValue, row, column: Variabilele care stochează valoarea veche a celulei, noua valoare a celulei, rândul și coloana în care se află celula.
* public StackObject(int oldValue, int newValue, int row, int column): Constructorul clasei "StackObject" care primește valorile și poziția celulei și le stochează în variabilele respective.
* public int getColumn(): Metoda care returnează coloana în care se află celula.
* public int getRow(): Metoda care returnează rândul în care se află celula.
* public int getNewValue(): Metoda care returnează noua valoare a celulei.
* public int getOldValue(): Metoda care returnează valoarea veche a celulei.

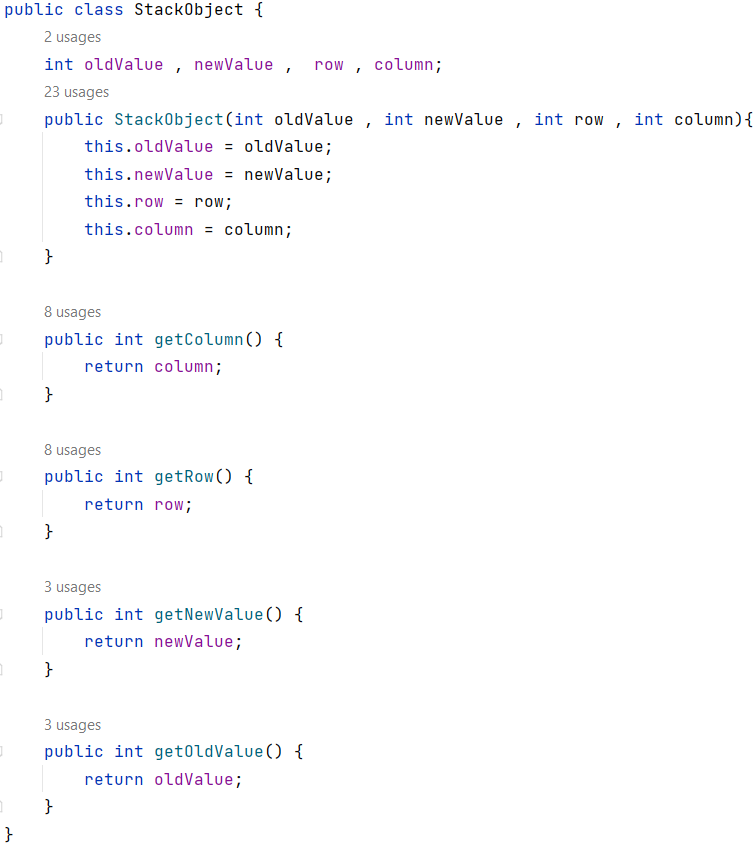


Figure 33

## 4.4 Testarea aplicației Mathdoku

Testarea Mathdoku este un proces esențial pentru a verifica funcționalitatea corectă și performanța aplicației. În timpul testării, sunt evaluate diferite aspecte ale aplicației pentru a identifica erori, comportamente neașteptate și pentru a verifica dacă funcționalitățile implementate funcționează conform specificațiilor și cerințelor.

Prin folosirea Application Lifecycle Management (ALM) am realizat 4 teste pentru a verifica funcționalitatea aplicației.

1. Testul pentru verificarea butoanelor care schimba mărimea fontului

Aplicația este deschisă in mod implicit cu marimea fontului setată pe mediu.

Urmărind pașii de testare schimbam fontul in ordinea următoare:

mic->mediu->mare

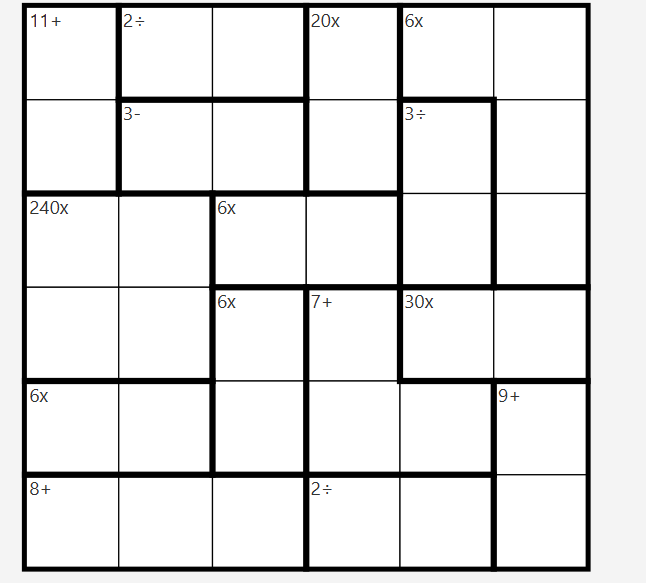


Figure 34

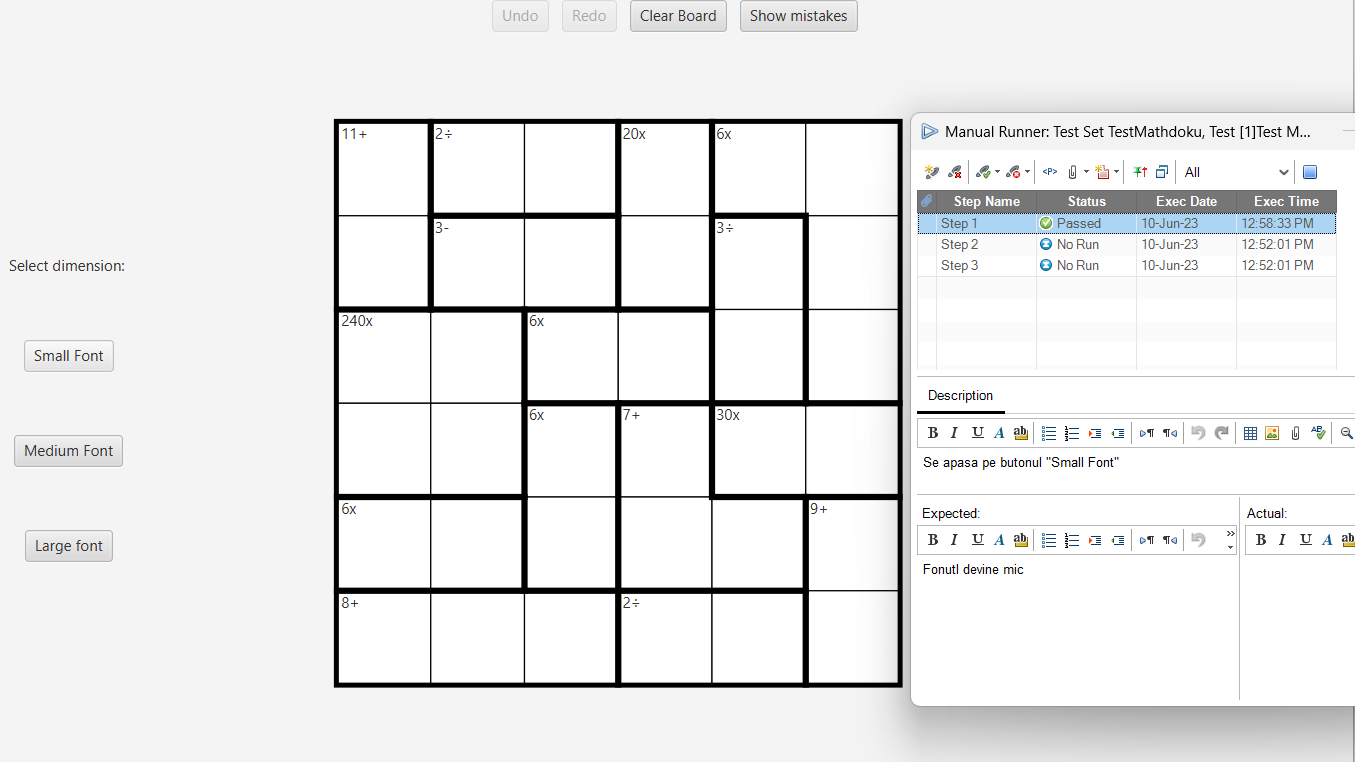


Figure 35

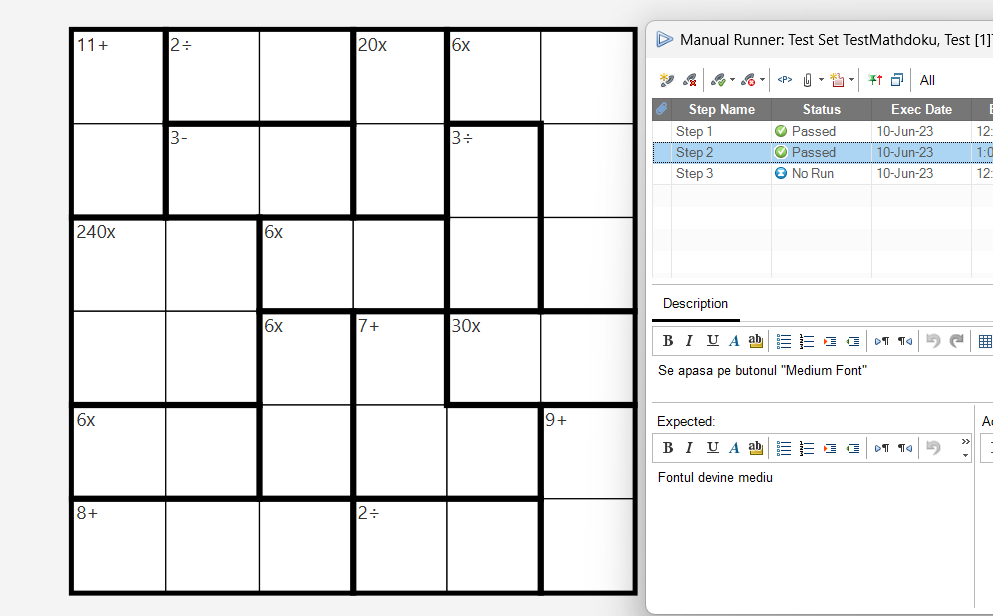


Figure 36

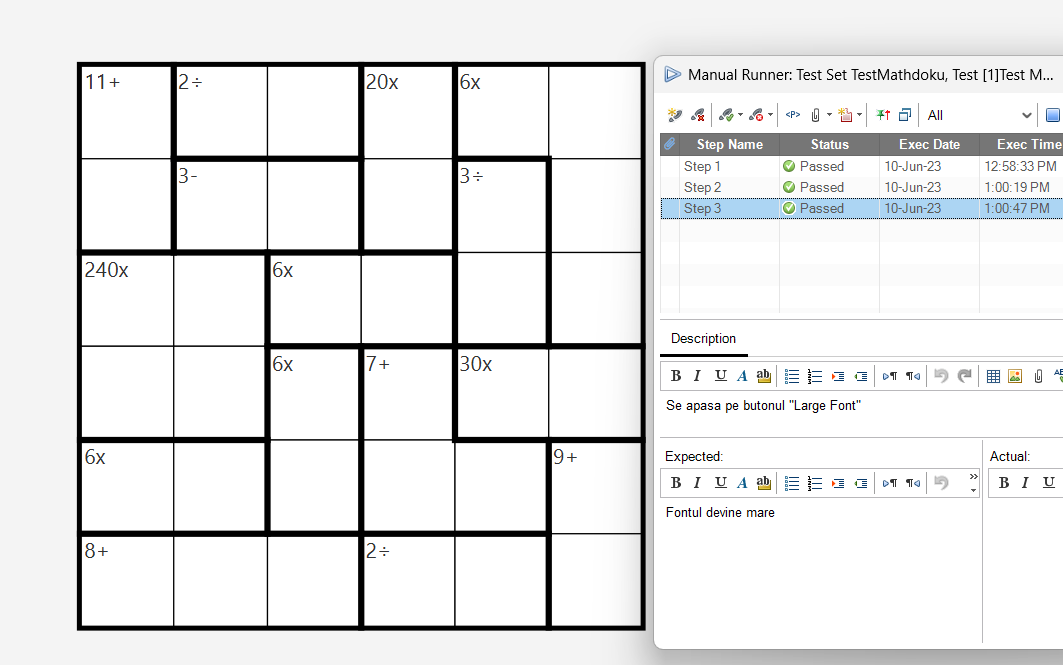


Figure 37

1. Testul pentru schimbarea dificultății jocului.

Aplicația este deschisa in mod implicit cu o tabla de joc 6x6.

Utilizatorul poate alege una dintre dificultățile mai ușoare sau, după preferințe una mai grea.

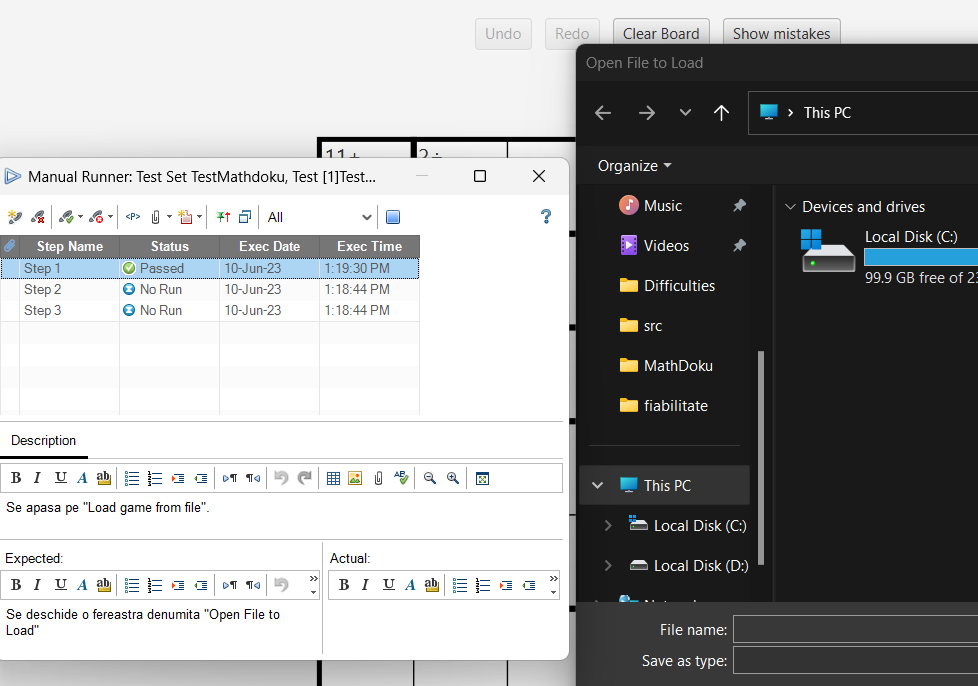


Figure 38

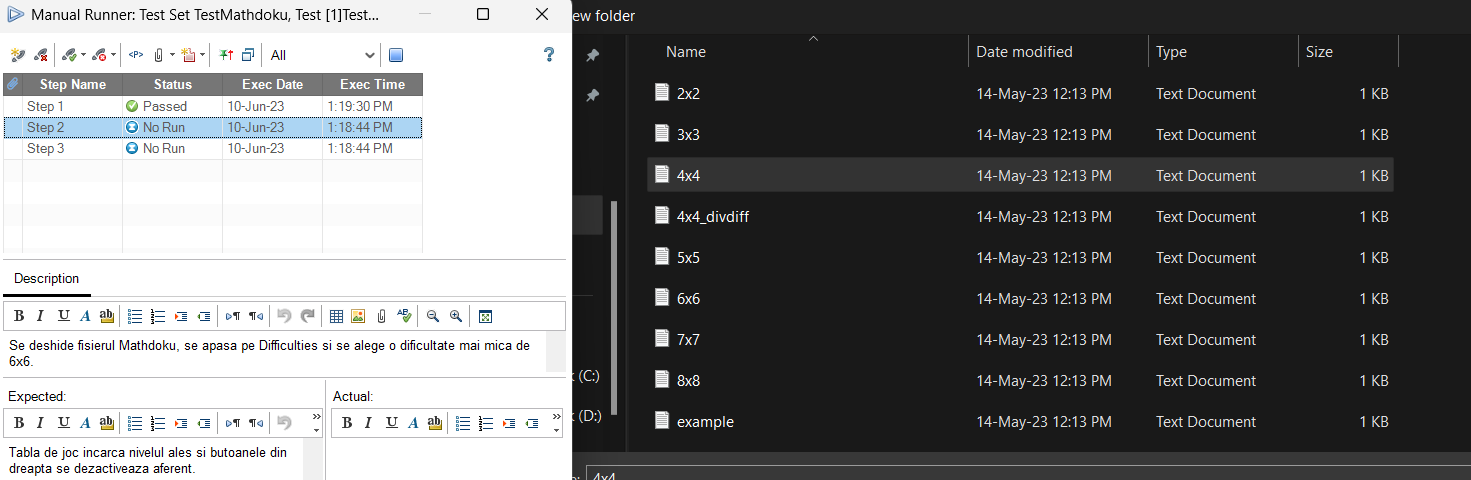


Figure 39

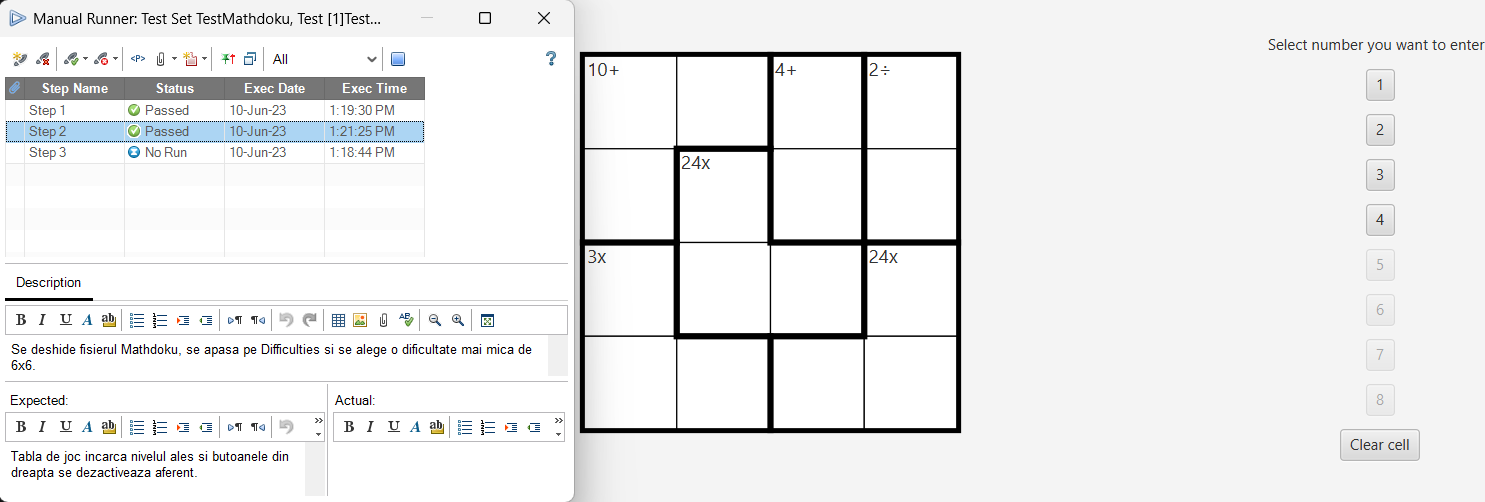


Figure 40

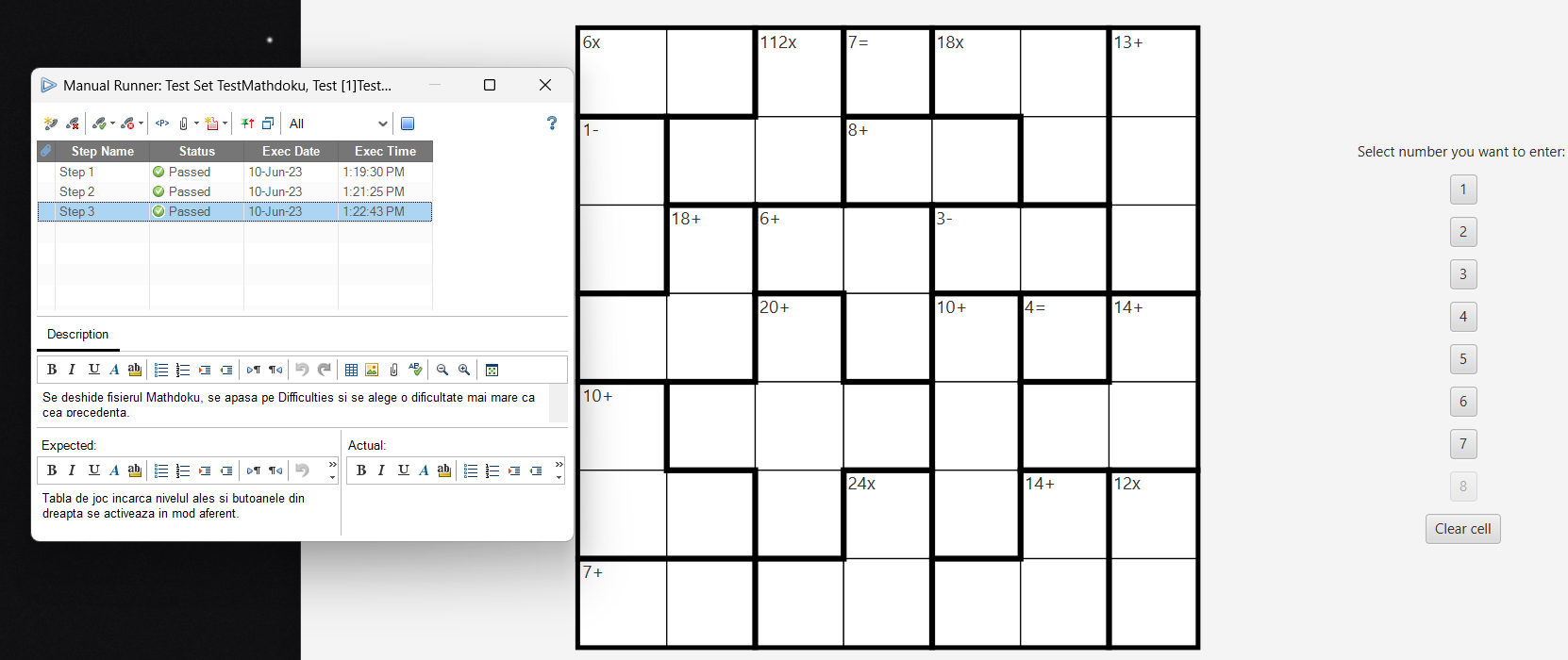


Figure 41

1. Testul pentru verificarea funcționalității butoanelor “Show mistakes” și “Solve!”

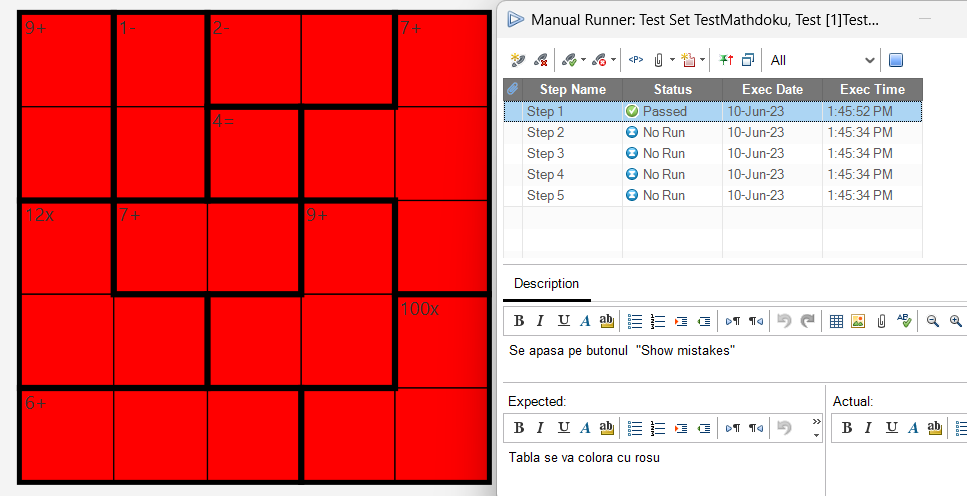


Figure 42

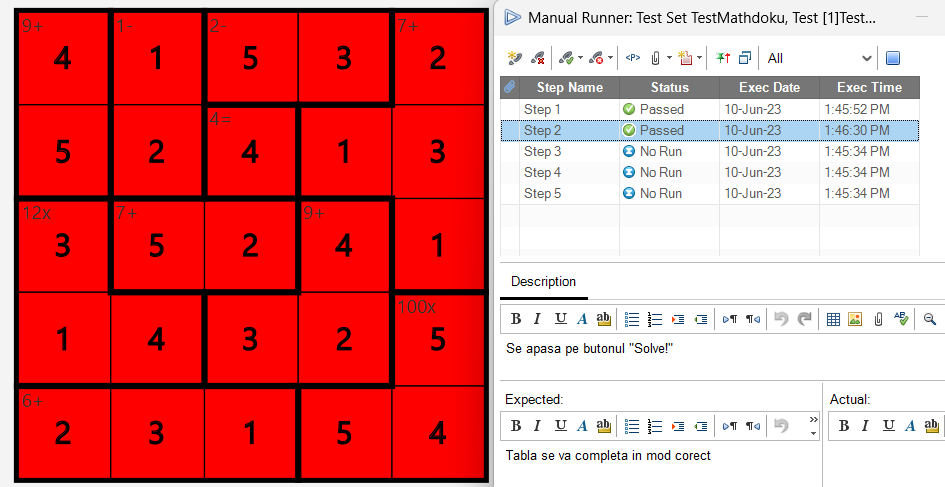


Figure 43

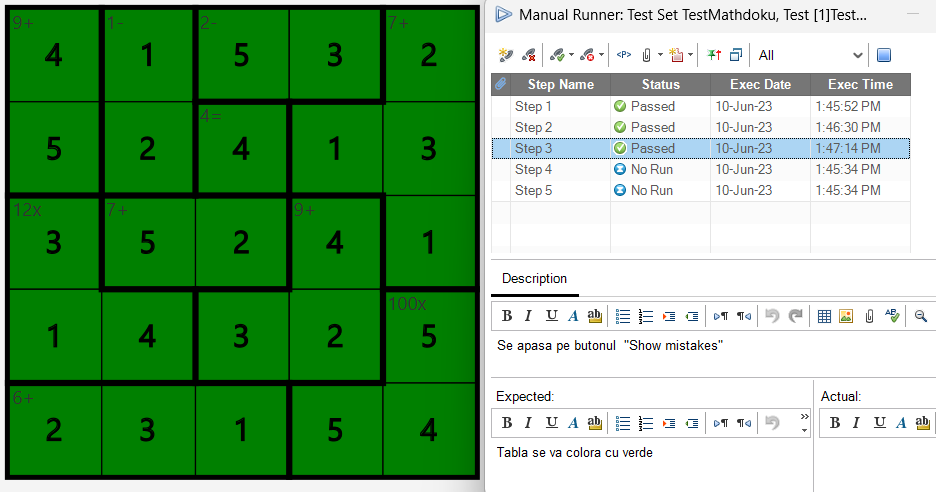


Figure 44

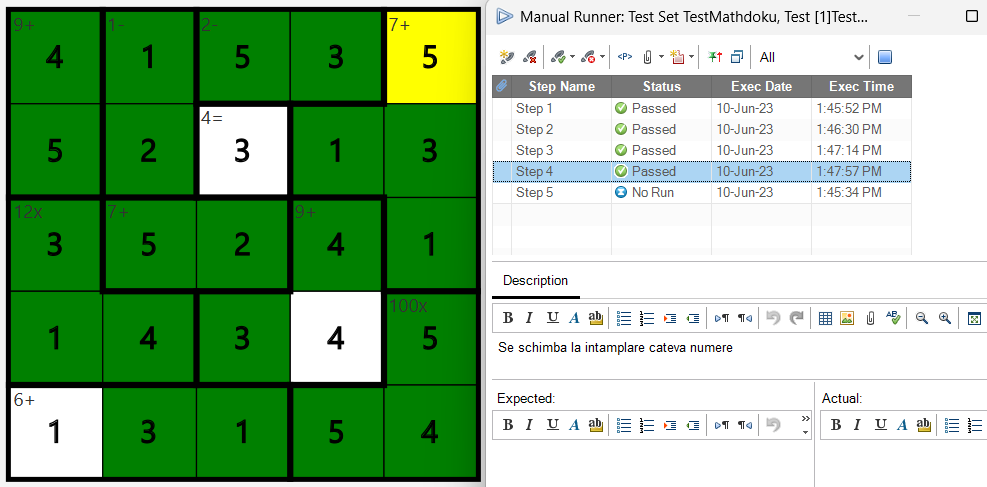


Figure 45

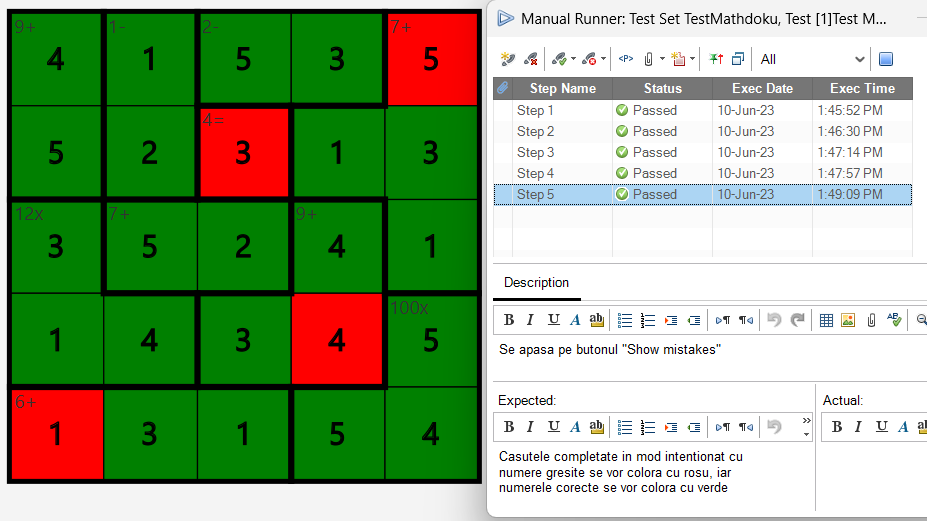


Figure 46

1. Testul pentru verificarea funcționalității butoanelor “Undo” și “Redo”

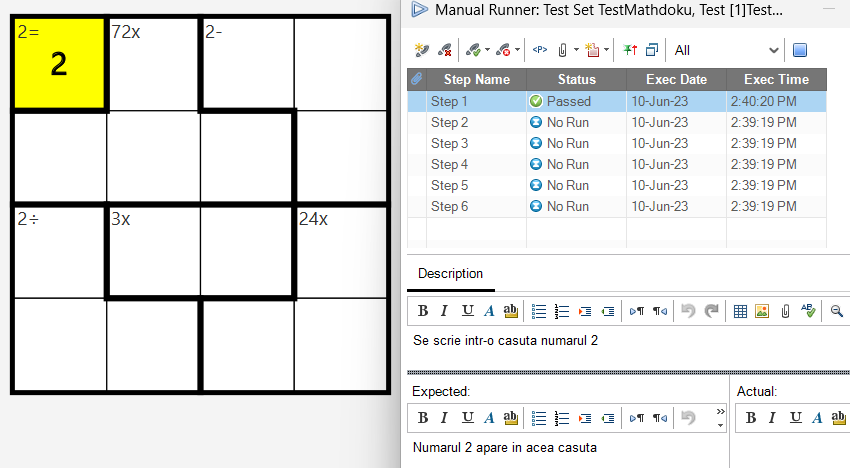


Figure 47

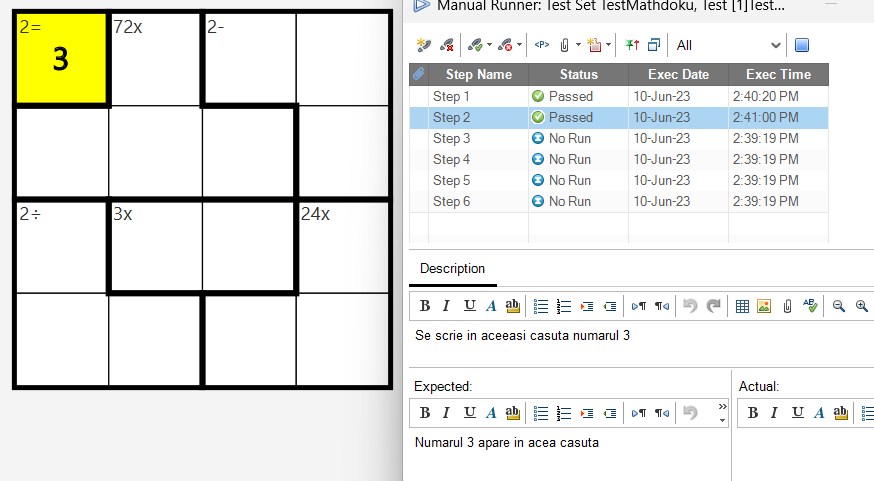


Figure 48

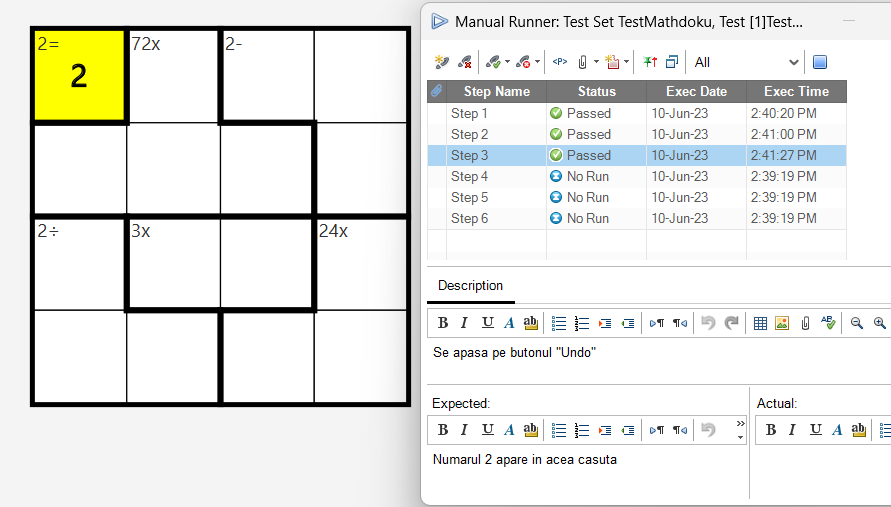


Figure 49

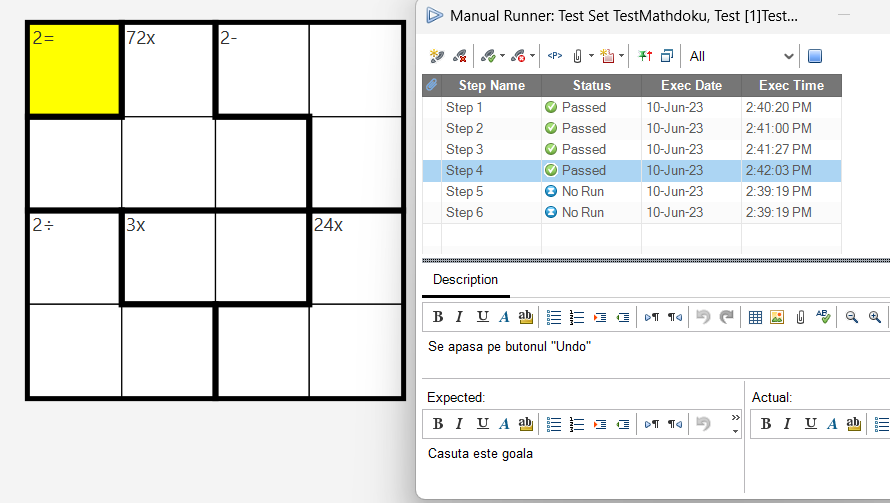


Figure 50

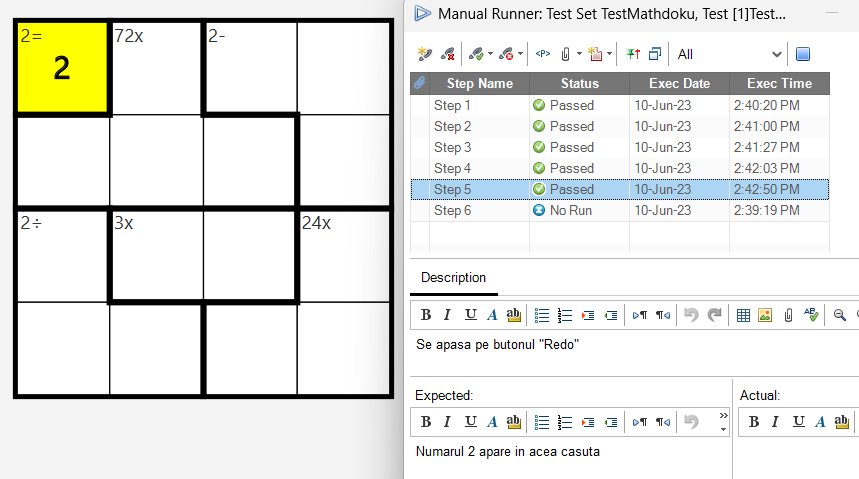


Figure 51

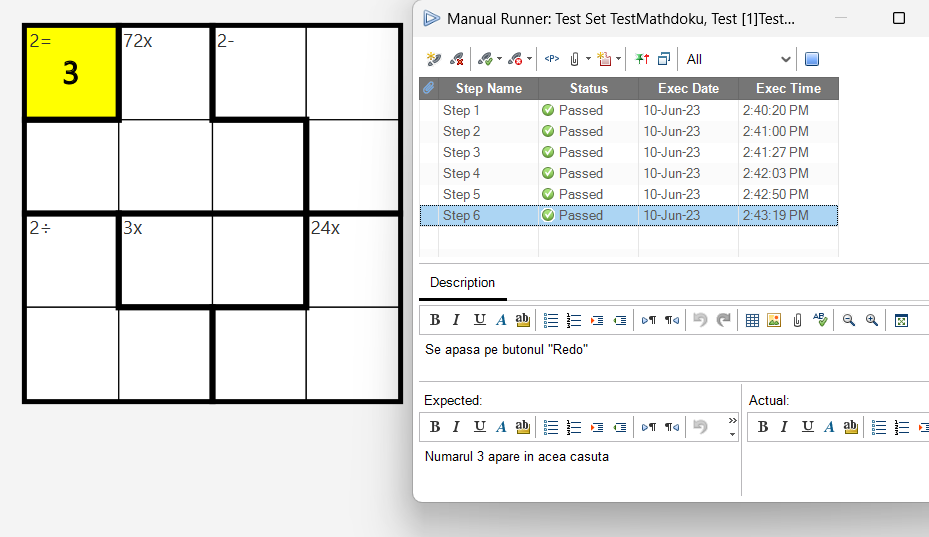


Figure 52

# Capitolul 5. Concluzii

Capitolul 5 marchează încheierea proiectului Mathdoku și oferă o perspectivă asupra rezultatelor obținute, a lecțiilor învățate și a direcțiilor recomandate pentru dezvoltarea ulterioară a aplicației. În acest capitol, se vor evidenția principalele concluzii trase din implementarea și testarea aplicației, se vor oferi recomandări pentru îmbunătățiri și se vor discuta posibile direcții viitoare de dezvoltare.

## 5.1 Rezultatele obținute

În această lucrare, am abordat dezvoltarea aplicației Mathdoku, un joc matematic captivant și stimulativ. Am explorat originea și evoluția Mathdoku, am analizat regulile și variantele acestuia și am examinat beneficiile rezolvării de puzzle-uri Mathdoku. Am utilizat tehnologii precum JavaFX și IntelliJ pentru dezvoltarea aplicației, având ca scop crearea unei interfețe intuitive și plăcute.

Prin intermediul procesului de analiză și proiectare, am identificat cerințele funcționale și non-funcționale ale aplicației Mathdoku și am proiectat o arhitectură modulară și o interfață utilizator atrăgătoare. Am dezvoltat aplicația Mathdoku cu succes, permițând utilizatorilor să rezolve puzzle-uri matematice în mod interactiv și captivant.

## 5.2 Concluzii generale

Dezvoltarea aplicației Mathdoku a oferit o experiență valoroasă în înțelegerea și aplicarea conceptelor matematice într-un context distractiv și interactiv. Jocurile matematice, cum ar fi Mathdoku, pot fi instrumente eficiente pentru dezvoltarea gândirii critice, a abilităților de rezolvare a problemelor și a concentrării.

Aplicația Mathdoku demonstrează beneficiile utilizării tehnologiei în promovarea educației și distracției simultane. Utilizatorii pot îmbunătăți abilitățile lor matematice într-un mod plăcut și relaxant, fără presiunea și monotonul metodelor tradiționale de învățare.

În viitor, aplicația Mathdoku poate fi îmbunătățită prin adăugarea de caracteristici suplimentare, cum ar fi un mod online pe baza de leaderboard, funcționalitatea de generare a puzzle-urilor într-un mod divers și integrarea cu alte platforme și dispozitive.

În final, dezvoltarea aplicației Mathdoku a reprezentat o provocare stimulativă și o oportunitate de a explora intersecția dintre matematică, tehnologie și divertisment.

# Bibliografie

1. Miyamoto, Tetsuya. "The Origin of KenKen." KenKen Puzzle LLC, 2008.
2. Shortz, Will. "The Worldwide Phenomenon of KenKen." The New York Times, 2008. [Articol despre popularizarea Mathdoku/KenKen în Statele Unite]
3. Sinclair, Ian. "Mathdoku: The History and Evolution of a Mathematical Puzzle." Mathematical Association, 2015. [Studiu despre istoria și evoluția Mathdoku]
4. Hewlett, Katherine. "Exploring the Mathematics Behind KenKen Puzzles." Mathematics Teaching in the Middle School, vol. 18, no. 4, 2012. [Articol despre aspectele matematice ale Mathdoku/KenKen]
5. Shortz, Will. "KenKen: For Experts Only." St. Martin's Griffin, 2009. [Colecție de puzzle-uri Mathdoku/KenKen cu niveluri avansate de dificultate]
6. Math Puzzle Society. (2022). Mathdoku Solver Algorithms and Techniques.
7. Flanagan, D. (2019). Java in a Nutshell: A Desktop Quick Reference (7th Edition). O'Reilly Media.
8. Lea, D. (2000). Concurrent Programming in Java: Design Principles and Patterns. Addison-Wesley Professional.
9. Java wikipedia - https://en.wikipedia.org/wiki/Java\_(programming\_language)
10. JavaFx wikipedia - https://en.wikipedia.org/wiki/JavaFX
11. IntelliJ IDEA - https://en.wikipedia.org/wiki/IntelliJ\_IDEA