People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

M'Hamed Bougara University
Faculty of Science
Department of Computer Science



Bachelor's Thesis

For the Award of the Bachelor's Degree in Computer Science

Specialization: Information Systems

Design of a Simulation Platform for Startup Failure Scenarios and Risk Awareness

Prepared by:
Halimi Asmaa
Belaid Rahma
Hamidat Chifa
Harhad Abderahmane

Supervised by:
Boudaoud Sid Ahmed

Examined by:
Prof. Maouche Amin Riad

Academic Year 2025/2026

أسماء

كتب في نهاية سطورنا أننا نلنا ما صبرنا لأجله،
فالحمد لله حبا وشكرا وامتنانا، حمدا يليق بجلاله وعظيم فضله.

بسم الله الرحمن الرحيم

وقبل أن أخطو خطوتي الأولى في طريق الحياة،

أهدي هذا العمل المتواضع إلى كل من كان لي سندا وعونا، عربون وفاء وامتنان.

إلى نفسي... التي مشت بثبات في دروب لم تكن سهلة، لكنها تعثرت ونهضت،
وحملت في قلبها يقينا بأن معية الله تكفي، فكانت أول من صبر وآمن حتى تحقق هذا الإنجاز.

إلى من لا يقاس عطاؤهما و لايعوض وجودهما،
الى من كانا النور في حياتي و الدعاء الذي برافقني دائما :
ابي و امي حفظهما الله و اطال في عمرهما،
وجعل الجنة مثواهما، وكتب لي برهما كما ربياني صغيرا.

وإلى إخوتي سميرة، وشهرة، هديل، محمد، الذين قبل فيهم:
﴿سنشد عضدك بأخيك﴾

إخوتي الذين لم أصل لولا دعمهم وعطائهم بلا مقابل، فأنتم
قوتي وملاذي بعد الله، أدامكم الله سندا ثابتا لي.

وإلى عائلتي الممتدة: خالي كمال وأحمد، وخالتي حسينة، وحياة، وفاطمة،
الذين كان لهم أثر جميل في حياتي في كل مرحلة من طريقي.

وإلى صديقاتي المقربات: ميرا، دعاء، وأمينة،
اللواتي كن لي رفيقات درب، ودعما حقيقيا،
وكن دائما مصدر تشجيع وكلمة طيبة خففت عني تعب الطريق.

وإلى أستاذي المؤطر بوداود، الذي كان أكبر داعم لي بعد الله،
ولم يبخل علي وتوجهاته ونصائحه طوال هذا العمل،
فله مني كل الاحترام والتقدير.

وإلى شركائي في هذا العمل، أصحاب العزم والمثابرة:
شفاء، ورحمة، وعبد الرحمن،
الذين كان لهم دور كبير في إتمام هذا الجهد،
فلكم مني كل التقدير والامتنان.

وإلى كل من جمعني بهم القدر في درب الحياة،
وترك في قلبي أثرا جميلا ولو بكلمة أو دعاء..
لكم مني كل الامتنان والمحبة.

وفي الختام، أسأل الله أن يجعل هذا العمل خالصا لوجهه الكريم، وأن يبارك فيه.

رحمة

﴿وَآخِرُ دَعْوَاهُمْ أَنِ الْحَمْدُ للهِ رَبِّ الْعَالَمِينَ﴾

ما سلكنا البداياتِ إلا بتيسيرِه، وما بلغنا النهاياتِ إلا بتوفيقه، وما حققنا الغاياتِ إلا بفضله، فالحمدُ لله حبّاً ورضا على ما وهبَ وأعطى.

من قال: أنا لها... نللها، وأنا لها، وإن أبتْ رغمًا عنها أتيتُ بها.

لم تكن الطريقُ دائمًا كما تمنّيت، لكنها كانت مليئةً بالدروس والخُطى التي قادتني إلى هنا. وبين لحُظاتِ تعبٍ وأخري من الأملَ، وبين سعي ودعاء، مضت الرحلةُ بهدوء أيّامها وتفاصيلها، حتى بلغتُ اليومَ ما كان بالأمس هدفًا أسعى إليه.

وما كان الوصولُ وليد لحظةٍ عابرة، بلَ كان ثمرة أيّام مضت بين اجتهادٍ وطموح، وبين تعثُرٍ أعقبه نهوض، حتى صار ما كان يومًا حلمًا، واقعًا أقفُ عنده اليومَ بامتنانٍ وفخر.

إلى أوّلُ من همس في أذني تراتيل القرآن، من كلّلَه الله بالهيبة والوقار، إلى الذي بنى ابتسامتي على سبيلِ شقائه، وسعى بكدِ يديه ليُوصلني إلى ما أُنا عليه اليوم. لم يكن تعبكَ خفيًّا، ولم يكن جهدك عابرًا، بل كان محفورًا في كلّ خُطوةٍ من طريقي، فكلّ خُطوةٍ خطوتُها كانت على دربٍ مهدّته لي بحبٍ وعطاء. أبي الحبيب، لك من القلب دعاءً لا ينقطع، وامتنانٍ لا تصفه الكلمات.

إلى من في كلّ نجاح أصلُ إليه أثرٌ من دعائها، وفي كلّ خُطوةٍ أخطوها بصمةٌ من عطائها. ولو منتحتُ الكلماتِ أبلَغَ معانيها، والعباراتِ أحملَ صورها، فلَن أوفيك حقّك. كنتِ النورَ الذي يُضيءُ طريقي، والسندَ الذي لا يميل، والدفءَ الذي بَيعثُ الطمأنينةَ في قلبي. حفظك الله لي نعمةً لا تزول.

إلى ضلعي الثابتِ، وأمانِ أيّامي، وسند العمرِ الذي لا يميل.
إلى من اشتدَّ بهم عضدي، واستقامت بهم خُطاي، فكانوا لي في الشدائدِ قوةٍ، وفي العثراتِ عونًا، وفي أيّام التعبِ ينابيعَ أرتوي منها صبرًا وأملاً.
اخي و اخواتي و ابنائهم،

إلى رفيقاتِ الدربِ،شكراً لقلوبٍ كانت سنداً في أيّام جمعتنا بين تعبٍ وأمل، ولذكرياتٍ جميلة خففت مشقّةَ الطريق، ولرفقة طيّبةٍ كانت خيرَ معين.
أسألُ الله أنِ يباركَ خُطاكُنَ، ويكتب لكُنَ التوفيق حيث اتجهتُنَ، وأن يديمَ بيننا المَودّة والمحبةَ.

وإلى أستاذي المشرف: بوداود سيد أحمد،
أتقدّم بجزيل الشكرِ وعظيم الامتنانِ، على كريم توجيهاته، وصادقِ دعمه، وما قدّمهُ منِ علم ونضج كانِ لهَ بالغُ الأثر في إنجاز هذا العمل.
فجزاه الله خيرَ الجزاء، وباركَ له علمه وعطاءه.

وفي الختام، أهدي هذا العمل لِكلّ من كان جزءًا من هذه الرحلة، ولِكلّ قلب منحني دعمًا، ولِكلّ يدٍ امتدّت بالعونَ، ولكلّ دعوةٍ صادقةٍ كانت سببًا في بلوغ هذه اللحظة.

شفاء

ليس من يقطع طريقاً بطّل إنما من يتقي الله البطل ، اطلب العلم ولا تكسل فما أبعد الخير عن أهل الكسل وسنين الجهد ان طالت ستطوى ، لها أمدٌ و للأمد انقضاء لنا بالله آمال و سلوى و عند الله ما خاب الرجاء ، إذا اشتدّت رياح اليأس فينا سيعقب ضيق شدتها الرخاء . اما بعد :

أستحضر مشهدي الأول في هذا المكان حين كانت الدموع تملأ عيني و خاطري يقول لا المكان مكاني ولا الأحلام أحلامي لكن اليوم أحمد الله جل في علاه فإن حلاوة الأقدار في الرضا و و الله أني رضيت ،فالحمد الله الذي يفضله تدرك الغايات و بعونه تذلل الصعوبات و بتوفيقه تحقق الأمنيات فما كان من خير فمن الله و ما سلكنا البدايات إلا بتيسيره و ما بلغنا النهايات إلا بتوفيقه و ما حققنا الغايات إلا بفضله .

أقف اليوم حاملة عهدا أن أظل وفية لكل قطرة دم سالت في سبيل القضية ، فإن أرض العزة لا تحتاج لجيل يفاخر بالشهادات فحسب بل إلى جيل يكتب و يبني و يعمر بقيمه و انتمائه قبل علمه.أهدي هذا العمل بكل تقدير و دعاء إلى اخواننا الصامدين ، المعتقلين و المرابطين في القطاع فاللهم إجعلنا لهم عونا لا عبئاً على هذه الأرض و استخدمنا لنصرة دين الحق .

أهدي تخرجي هذا أولا لنفسي : إذ لا أراك اليوم مجرد خريجة إنما إنسانة انتزعت حقها في الفرح من بين أنياب التعب ،فخورة بك ان جعلت من الصبر حقل سنابل و من العناء سلّماً نحو العلا .

إلى من أكل التعب من عمره ليشتري لي مستقبلا ، إلى من حصد الأشواك عن دربي ليمهد لي طريق العلم أبي الغالي ،إلى منبع النور و ملاذ الروح و مستودع الصبر إلى مدرسة حياتي و أعظم أستاذة عرفتها الأيام إلى أمي حبيبتي تخرجي هذا لم يكن ليكن لولا دعواتك فلك مني قبلة اعتذار عن كل لحظة قصرت فيها ، إلى التي حضنتني بقلبها قبل يديها الى دليل القلب في رحم التعثرات أختي و أمي الثانية أنفال ، إلى الأيادي التي استندت عليها دائما أخي و أخواتي و بناتهم و أولادهم إضافّة لرفيقات دربي ادعوا الله ان أكون عند حسن ظنكم بي ، إلى ذهبي الخالص إلى من كانا المُتكأ جدتي و خالتي رحمكما الله و رزقكما جنة الفردوس الأعلى ، إلى دعوتي المستجابة بالصحبة الصالحة إلى أنسي التي تجعل من العتمة ممرا إجباريا نحو الضوء حفظنا الله لبعضنا

كمّا أتقدم بالشكر الجزيل للمشرف بوداود. س على توجيهاته فلو أمسيت بين الناس بدرا فلولا الشمس ما سطع الهلال و لا أنسى زميلاتي في المشروع فالشكر أدنى ما أقدمه لهم ،أستودعكم الله و السلام عليكم و رحمة الله تعالى وبركاته .



عبـد الرحمـان

الحمد لله الذي نفتتح الكلام والحمد لله فالق الأنوار، جاعل الليل والنهار
وخالق السماوات بلا عمد ورزق الرزق ولم ينس أحد،
ثم الصلاة والسلام على سيدنا محمد المختار صلى الله عليه وسلم

إلى التي حملتني وهنا على وهن التي أهدتها الحياة التعب والحرمان،
إلى التي أرضعتني وسقتني من ماء الحياة.....
إلى نور عيني وضياء قلبي أمي الغالية
" أمينه ربعي "

إلى الذي تكبد الشدائد وكان عرق جبينه منير دربي،
إلى من إشترى لي أول قلم ودفعني
بكل ثقة على خوض الصعاب، إلى من كلله الله بالهيبة والوقار.
إلى من علمني العطاء بدون إنتظار،
إلى من أحمل إسمه بكل إفتخار أبي العزيز
" مختار حرحاد "

وإلى من ترعرعت معهم وزاد حبي لهم ونما غصني بينهم،
إلى رياحين حياتي.... إخوتي
عماد و ياسين

إلى جدتي باية
إلى خالاتي سهام و وهيبة
إلى كل الأهل والأقارب

كما أهدي هذا العمل المتواضع
إلى أصدقائي : ياسر و بلال

وإلى من شاركوني خطوات مذا المتوار خطوةً بخطوة،
من تقاسمنا معاً عناء البحث وحلاوة الإنجاز،
إلى رفيقات الدرب وشريكات العلم والفكر...
زميلاتي في هذا المشروع
أسماء، رحمة و شفاء
وإلى كل من قاسمتي وساعدني في إنجاز هذه المذكرة

Acknowledgment

First of all, we thank Allah for giving us the strength, patience, and ability to accomplish this work and successfully complete our project.

We would like to express our sincere gratitude to our supervisor, **Dr. Boudaoud**, for his valuable guidance, continuous support, encouragement, and patience throughout this project. His advice and availability were greatly appreciated and played an important role in the completion of this work.

We would like to thank Mr Maouche for his acceptance to evaluate our modest work and give us his valuable feedback and recommendations to make this project more useful for our country's business men.

We also extend our thanks to all our professors for their efforts, knowledge, and support during these three years of study. Their teaching and guidance contributed greatly to our academic development.

Special thanks go to our classmates and families for their continuous encouragement, motivation, and support.

Finally, we would like to thank everyone who contributed, directly or indirectly, to the success of this work.

Abstract

Startups operate in highly uncertain and dynamic environments where financial instability, poor strategic decisions, market changes, and operational difficulties can quickly lead to failure. Understanding these risks is essential for improving entrepreneurial decision-making and increasing awareness of startup challenges.

This project presents the design and implementation of an interactive simulation platform dedicated to startup failure scenarios and risk awareness. The proposed system combines simulation techniques, rule-based logic, and artificial intelligence methods to model startup evolution and evaluate the impact of user decisions over time.

The platform is based on a client-server architecture composed of a React and TypeScript front-end, a FastAPI back-end, and a MySQL database. It provides three simulation modes: rule-based simulation, AI-based prediction, and a hybrid approach combining both methods. The system also integrates Explainable Artificial Intelligence (XAI) mechanisms to improve the interpretability of prediction results.

To evaluate the intelligent component of the platform, several machine learning models including Decision Tree, Random Forest, and Logistic Regression were tested and compared using performance and efficiency metrics. Experimental results showed that the Decision Tree model achieved the best balance between prediction accuracy, computational efficiency, and interpretability.

The developed platform offers users an educational and analytical environment where they can explore startup risk factors, simulate different business scenarios, and better understand the mechanisms that may lead to startup failure.

Keywords: Startup Failure, Simulation Platform, Artificial Intelligence, Machine Learning, Risk Analysis, Decision Support Systems, Explainable AI.

Résumé

Les startups évoluent dans des environnements fortement incertains et dynamiques, où l'instabilité financière, les mauvaises décisions stratégiques, les changements du marché et les difficultés opérationnelles peuvent rapidement conduire à l'échec. Comprendre ces risques est essentiel afin d'améliorer la prise de décision entrepreneuriale et de sensibiliser aux défis liés aux startups.

Ce projet présente la conception et l'implémentation d'une plateforme interactive de simulation dédiée aux scénarios d'échec des startups et à la sensibilisation aux risques entrepreneuriaux. Le système proposé combine des techniques de simulation, une logique basée sur des règles, ainsi que des méthodes d'intelligence artificielle afin de modéliser l'évolution d'une startup et d'évaluer l'impact des décisions prises par l'utilisateur au fil du temps.

La plateforme repose sur une architecture client-serveur composée d'un front-end développé avec React et TypeScript, d'un back-end utilisant FastAPI, ainsi que d'une base de données MySQL. Elle propose trois modes de simulation : une simulation basée sur des règles, une prédiction basée sur l'intelligence artificielle et une approche hybride combinant les deux méthodes. Le système intègre également des mécanismes d'Intelligence Artificielle Explicable (XAI) afin d'améliorer l'interprétabilité des résultats prédictifs.

Pour évaluer la partie intelligente de la plateforme, plusieurs modèles de machine learning, notamment Decision Tree, Random Forest et Logistic Regression, ont été testés et comparés à l'aide de métriques de performance et d'efficacité. Les résultats expérimentaux ont montré que le modèle Decision Tree offre le meilleur compromis entre précision prédictive, efficacité computationnelle et interprétabilité.

La plateforme développée fournit ainsi un environnement pédagogique et analytique permettant aux utilisateurs d'explorer les facteurs de risque des startups, de simuler différents scénarios entrepreneuriaux et de mieux comprendre les mécanismes pouvant conduire à l'échec d'une startup.

Mots-clés : Échec des startups, Plateforme de simulation, Intelligence artificielle, Machine Learning, Analyse des risques, Systèmes d'aide à la décision, Intelligence Artificielle Explicable.

Contents

DedicationI **I**

DedicationII **I**

DedicationIII **II**

DedicationIV **IV**

Acknowledgement **V**

Abstract **VI**

Résumé **VII**

List of Figures **X**

List of Tables **XI**

General Introduction **1**

1 State of the Art **4**

1.1 Introduction 4

1.2 Taxonomy of Startup Failure Causes 4

1.2.1 Financial Causes 4

1.2.2 Market-Related Causes 5

1.2.3 Strategic Causes 5

1.2.4 Organizational Causes 5

1.2.5 External and Environmental Causes 5

1.3 Systemic Approach to Startup Failure 5

1.4 Modeling Startup Risk 5

1.5 Decision-Making Approaches in Simulation 6

1.5.1 Rule-Based Systems 6

1.5.2 Machine Learning Approaches 6

1.5.3 Hybrid Approaches 9

1.6 Simulation Systems and Dynamic Modeling 9
1.7 Synthesis 10

2 Study of Existing Solutions **11**
2.1 Introduction 11
2.2 Business Simulation Platforms 11
2.3 Startup-Oriented Platforms 12
2.4 Decision Support Systems 12
2.5 Simulation and Artificial Intelligence Approaches 12
2.6 Limitations of Existing Solutions 13
2.7 Comparative Analysis 13
2.8 Conclusion 14

3 System Design **15**
3.1 Introduction 15
3.2 System Architecture 15
3.2.1 User and Presentation Layer (Front-end) 16
3.2.2 Logic and Decision Layer (Back-end) 16
3.3 Database Design 16
3.3.1 Conceptual Data Model (CDM) 17
3.3.2 Logical Data Model (LDM) 17
3.4 Sequence Diagrams 18
3.4.1 Registration Diagram 18
3.4.2 Authentication Diagram 19
3.4.3 Global Simulation Sequence Diagram 19
3.5 Activity Diagram 20
3.6 Conclusion 21

4 System Implementation **22**
4.1 Introduction 22
4.2 Technologies Used 22
4.3 User Interfaces 23
4.3.1 Login Interface 23
4.3.2 Simulation Interface 23
4.3.3 Dashboard Interface 24
4.3.4 AI Prediction Result Interface 24
4.4 Machine Learning Evaluation 25
4.5 Dataset Overview 25
4.6 Machine Learning Models 26
4.6.1 Decision Tree 26

4.6.2 Random Forest 26
4.6.3 Logistic Regression 26
4.7 Experimental Results 27
4.8 Efficiency Analysis 28
4.9 Best Model Selection 29
4.10 conclusion 29

General Conclusion **30**

List of Figures

3.1 Global System Architecture 15
3.2 Conceptual Data Model (CDM) 17
3.3 Relational Schema of the Logical Data Model (LDM) 18
3.4 Registration Sequence Diagram 19
3.5 Authentication Sequence Diagram 19
3.6 Global Simulation Sequence Diagram 20
3.7 Global Activity Diagram of the Startup Failure Simulation Platform 21

4.1 Login Interface 23
4.2 methods 24
4.3 variables 24
4.4 Dashboard Interface 24
4.5 AI-Based Simulation Result Interface 25
4.6 ML Model Performance Comparison 27
4.7 Computational Performance Comparison 28

List of Tables

1.1 Comparison of machine learning algorithms for startup failure prediction . 10

2.1 Comparison of Existing Platforms 14

4.1 Model Performance Comparison 27
4.2 Computational Performance Comparison 28

General Introduction

In recent years, startups have emerged as key drivers of innovation, technological progress, and economic transformation. By introducing new business models and disruptive solutions, startups contribute significantly to competitiveness and job creation. However, despite their importance, startups operate in highly uncertain and dynamic environments, where failure is not an exception but rather a structural characteristic. Empirical studies indicate that a large proportion of startups fail during their early stages due to a combination of financial, strategic, and market-related factors [7, 25].

Globally, the scale of startup failure is striking. Approximately 90% of startups fail overall, with 10% collapsing within the first year and 70% during years two through five. Less than half of all startups survive the first five years, and over two-thirds fail within ten years. According to CB Insights, which analyzed post-mortems of over 100 failed startups, 42% cited 'no market need' as a primary cause of failure, while 70% ran out of capital—though this is often a final symptom rather than the root cause.

In Algeria, the situation is particularly challenging. Despite multiple state-initiated reforms, startup failure rates remain persistently high. The ecosystem ranks 109th globally with 52 identified startups, yet only 15% of labeling applications are successful, and less than 8% of startups have secured contracts with local enterprises. Academic research confirms that failures predominantly result from deficient managerial competencies, inadequate strategic planning, financing accessibility challenges, bureaucratic impediments, and weak integration between research systems and market demands.

The causes of startup failure are multifaceted and interconnected. Financial instability, characterized by poor cash flow management and insufficient funding, remains one of the primary causes of failure. Classical models such as those proposed by Altman and Ohlson have demonstrated the importance of financial indicators in predicting business failure [1, 21]. In addition to financial aspects, market-related issues such as the lack of product-market fit play a critical role. As emphasized in the Lean Startup methodology, the ability to rapidly test hypotheses and adapt to customer feedback is essential for survival in uncertain environments [24].

Beyond individual factors, startup failure must be understood as a systemic phenomenon resulting from complex interactions between multiple variables. System dynamics approaches, as introduced by Sterman, highlight the importance of feedback loops,

delays, and nonlinear relationships in organizational behavior [28]. In such systems, small variations in one variable can propagate and amplify through the system, leading to significant consequences. This perspective is particularly relevant for modeling startup evolution, where decisions related to financing, strategy, and operations are interdependent and evolve over time.

Traditional approaches to entrepreneurship education and decision-making are often limited in their ability to capture this complexity. Entrepreneurs typically rely on theoretical knowledge, case studies, or real-world experience, where learning from failure can be costly and irreversible. In this context, simulation-based approaches offer a powerful alternative by enabling experimentation in a controlled and risk-free environment. Business simulation systems have been widely used for training and decision support, allowing users to explore different strategies and observe their outcomes without real-world consequences [9].

The integration of artificial intelligence techniques further enhances the capabilities of simulation platforms. Machine learning methods allow systems to learn from data and improve predictive performance over time [19, 2]. In particular, algorithms such as Random Forests have demonstrated strong performance in classification and prediction tasks, including failure prediction [3]. Reinforcement learning approaches extend this capability by enabling agents to learn optimal decision-making strategies through interaction with the environment [29]. These methods are particularly suitable for modeling sequential decision processes under uncertainty.

In parallel, rule-based systems, rooted in expert systems and artificial intelligence research, provide a complementary approach by encoding domain knowledge in the form of explicit rules [12, 20]. While these systems offer high interpretability and control, they may struggle to capture complex and dynamic interactions. Therefore, combining rule-based reasoning with data-driven approaches can lead to more robust and flexible simulation models.

The main problematic addressed in this project is how to design a simulation platform capable of modeling startup failure scenarios while integrating both systemic dynamics and intelligent decision-making mechanisms. Such a platform must capture the multidimensional nature of startup environments, represent the interactions between key variables, and provide meaningful feedback to users. It should also allow the comparison of different decision-making approaches, including rule-based systems and artificial intelligence techniques.

The objective of this project is therefore to design and implement an interactive simulation platform for startup failure scenarios and risk awareness. The platform aims to transform theoretical concepts into measurable variables, simulate their evolution over time, and evaluate the impact of user decisions. By doing so, it provides a pedagogical and analytical tool that helps users understand the mechanisms of failure and improve

their strategic thinking.

This report is structured as follows. Chapter 1 presents the state of the art and introduces the theoretical and methodological foundations of the project, including the taxonomy of failure causes, systemic modeling, and decision-making approaches. Chapter 2 analyzes existing solutions and simulation platforms. Chapter 3 describes the design of the proposed system, including its architecture and modeling framework. Chapter 4 presents the implementation and experimental results. Finally, a general conclusion summarizes the main contributions and outlines future research directions.

Chapter 1

State of the Art

1.1 Introduction

Startups play a central role in innovation and economic development by introducing new technologies and disruptive business models. Despite their importance, startups operate in highly uncertain environments where failure is frequent. Studies indicate that a large proportion of startups fail during their early stages [7, 27].

Understanding the mechanisms behind startup failure is essential for both researchers and practitioners. This chapter presents the theoretical foundations of startup failure and introduces the key concepts used in this project. It focuses on three main aspects: the identification of failure factors, the systemic modeling of these factors, and the decision-making approaches used in simulation systems.

1.2 Taxonomy of Startup Failure Causes

Startup failure is a multidimensional phenomenon resulting from the interaction of several factors. Based on academic literature and industry reports, failure causes can be grouped into five main categories.

1.2.1 Financial Causes

Financial issues are among the most critical reasons for startup failure. These include insufficient funding, poor cash flow management, and uncontrolled expenses. Classical models such as those proposed by Altman and Ohlson highlight the importance of financial indicators in predicting business failure [1, 21].

1.2.2 Market-Related Causes

A major cause of failure is the lack of product-market fit, meaning that the product does not meet a real market need. According to CB Insights, a significant proportion of startups fail due to insufficient demand [5]. The Lean Startup approach emphasizes the importance of rapid experimentation and customer feedback to address this issue [24].

1.2.3 Strategic Causes

Strategic mistakes include poor decision-making, delayed adaptation to market changes, and inappropriate timing. Startups that fail to pivot when necessary or grow too rapidly without proper control often encounter serious challenges.

1.2.4 Organizational Causes

The quality of the team and internal management plays a crucial role in startup success. Weak leadership, lack of coordination, and poor governance structures can lead to inefficient execution and reduced performance [25].

1.2.5 External and Environmental Causes

External factors such as regulatory constraints, economic crises, and market disruptions can also lead to failure. These factors are often beyond the control of startups but must be considered in risk analysis.

1.3 Systemic Approach to Startup Failure

Startup failure cannot be explained by isolated factors alone. It results from complex interactions between financial, market, strategic, and organizational variables. A systemic perspective is therefore necessary.

System dynamics theory considers organizations as dynamic systems characterized by feedback loops, delays, and nonlinear relationships [28]. In such systems, small changes in one variable can lead to significant effects over time.

For example, weak market demand may reduce revenue, which in turn affects financial stability and limits the ability to invest in product development or marketing. These interactions create feedback loops that can accelerate failure.

1.4 Modeling Startup Risk

To simulate startup behavior, theoretical concepts must be transformed into measurable elements. This involves identifying key variables that represent the state of the startup

and its environment.

These variables typically include financial indicators, market performance, team capabilities, and external risks. Their evolution over time reflects the impact of decisions made by the entrepreneur.

The objective is not to achieve perfect prediction, but to provide a realistic representation of startup dynamics that supports analysis and learning.

1.5 Decision-Making Approaches in Simulation

Simulation platforms rely on decision-making mechanisms to evaluate user actions and generate outcomes. Two main approaches are commonly used.

1.5.1 Rule-Based Systems

Rule-based systems apply predefined logical rules to determine system behavior. Each rule represents expert knowledge and defines how variables evolve under specific conditions [12, 20].

These systems offer high interpretability and allow precise control over the simulation process. However, they may struggle to capture complex and dynamic interactions.

1.5.2 Machine Learning Approaches

Machine learning techniques enable systems to learn patterns from data and improve performance over time [19, 2]. They are widely used for prediction and classification tasks.

Three fundamental algorithms are particularly relevant for modeling startup failure: logistic regression, decision trees, and random forests. Each offers distinct advantages and limitations that must be considered when designing simulation systems.

Logistic Regression

Logistic regression is a statistical method used for binary classification problems, where the outcome of interest can take one of two possible values [14]. In the context of startup failure prediction, this algorithm estimates the probability that a given startup will fail based on a set of explanatory variables. The core principle of logistic regression lies in modeling the log-odds of the event as a linear combination of the input features.

Mathematically, the logistic regression model can be expressed as:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$$

where p represents the probability of startup failure, β_0 is the intercept term, β_i are the coefficients associated with each predictor variable x_i, and n is the number of features.

The probability p is then obtained through the logistic (sigmoid) function:

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n)}}$$

This function maps any real-valued input to a value between 0 and 1, making it suitable for probability estimation. The coefficients β_i are typically estimated using maximum likelihood estimation, which finds the parameter values that maximize the likelihood of observing the actual outcomes in the training data [18].

In startup failure analysis, logistic regression offers several advantages. First, it provides interpretable coefficients that directly indicate the direction and magnitude of each variable's influence on the likelihood of failure. For instance, a positive coefficient for a financial ratio such as debt-to-equity would suggest that higher leverage increases the probability of failure. Second, the model is computationally efficient and requires relatively little training data compared to more complex algorithms. Third, logistic regression produces well-calibrated probability estimates, which are valuable for risk assessment and decision-making.

However, logistic regression has important limitations. The algorithm assumes a linear relationship between the log-odds and the predictor variables, which may not hold true in complex startup ecosystems where non-linearities and interactions are common. Additionally, logistic regression is sensitive to outliers and multicollinearity among features. Despite these drawbacks, the method remains a valuable baseline model for prediction tasks and provides a foundation for understanding more advanced approaches.

Decision Trees

Decision trees are non-parametric supervised learning algorithms that make predictions by recursively partitioning the input space based on the values of the features [22, 4]. This hierarchical structure resembles a flowchart where each internal node represents a test on a specific attribute, each branch corresponds to the outcome of the test, and each leaf node provides a final prediction (a class label for classification or a continuous value for regression).

The construction of a decision tree involves selecting the most informative feature to split the data at each node. For classification tasks, the quality of a split is typically measured using criteria such as Gini impurity or information gain (entropy reduction). The Gini impurity is defined as:

$$G = 1 - \sum_{k=1}^{K} p_k^2$$

where p_k is the proportion of instances belonging to class k in the node, and K is the

number of classes. A split that reduces impurity most effectively is preferred.

Information gain, based on Shannon entropy, is another commonly used criterion:

$$Entropy = -\sum_{k=1}^{K} p_k \log_2(p_k)$$

Information gain measures the reduction in entropy achieved by splitting the data according to a given feature.

For startup failure prediction, decision trees offer remarkable interpretability. The visual structure of the tree makes it easy to understand the decision path for any particular instance. For example, a simple decision rule might indicate: "If the monthly burn rate exceeds $50,000 and the product-market fit score is below 0.3, then the startup is classified as high-risk." Such rules can be directly communicated to entrepreneurs and investors, facilitating transparency and trust.

Decision trees are also capable of handling both numerical and categorical variables without requiring extensive preprocessing. They are inherently robust to outliers and irrelevant features, as the splitting process naturally ignores variables that do not contribute to reducing impurity.

However, decision trees suffer from significant limitations. The most critical issue is overfitting, where the tree becomes too complex and memorizes noise in the training data rather than learning general patterns. This problem is typically addressed through pruning techniques, which remove branches that do not provide statistically significant improvements in prediction accuracy.

Furthermore, decision trees are unstable: small variations in the training data can lead to completely different tree structures, making the model sensitive to random fluctuations. Despite these challenges, decision trees are widely used in practice due to their simplicity, interpretability, and ability to capture non-linear relationships.

Random Forest

Random forests represent an ensemble learning method that combines multiple decision trees to improve predictive performance and robustness [3]. The algorithm builds a large collection of decision trees, each trained on a different bootstrap sample of the original training data. Additionally, at each node split, only a random subset of the features is considered, introducing further diversity among the trees.

Formally, for a dataset with N samples and M features, the random forest algorithm proceeds as follows:

1. For each tree $t = 1$ to T, draw a bootstrap sample of size N from the training data.
2. At each node of the tree, randomly select m features from the total M features, where typically $m \approx \sqrt{M}$ for classification tasks.

3. Choose the best split among the m features according to a chosen splitting criterion.

4. Grow the tree to its full depth without pruning.

5. Repeat steps 1–4 for all T trees.

For classification, the final prediction is determined by majority voting among the T trees. For regression problems, the average of the individual predictions is used.

Random forests address the overfitting and instability problems of single decision trees through two mechanisms. First, bootstrap aggregating (bagging) reduces variance by averaging predictions over multiple models trained on different subsets of the data. Second, the random feature selection ensures that the trees are decorrelated, preventing the ensemble from relying too heavily on a single dominant feature.

In the context of startup failure prediction, random forests offer several compelling advantages. They typically achieve superior predictive accuracy compared to logistic regression and single decision trees, particularly in high-dimensional settings where complex interactions exist between variables. The algorithm can handle large feature spaces and automatically captures non-linear relationships and interactions without requiring explicit specification.

Moreover, random forests provide a natural measure of feature importance, which indicates the contribution of each variable to the model's predictive power. This information is valuable for identifying the most critical drivers of startup failure and can guide entrepreneurs in prioritizing their actions.

The limitations of random forests include reduced interpretability compared to single decision trees, as the ensemble of hundreds of trees is difficult to visualize and explain to non-technical stakeholders. Additionally, the algorithm requires more computational resources and memory than simpler methods. The model may also suffer from bias in scenarios where the data is highly imbalanced, although techniques such as balanced sampling or weighted voting can mitigate this issue.

The following table summarizes the key characteristics of the three algorithms:

1.5.3 Hybrid Approaches

Combining rule-based reasoning with machine learning techniques provides a balanced solution. Hybrid systems benefit from both interpretability and adaptability, making them particularly suitable for simulation platforms.

1.6 Simulation Systems and Dynamic Modeling

Simulation systems provide a controlled environment where users can experiment with decisions and observe their consequences without real-world risks. They are widely used

Table 1.1: Comparison of machine learning algorithms for startup failure prediction

Criterion	**Logistic Regression**	**Decision Tree**	**Random Forest**
Interpretability	High	Very High	Low
Non-linear relationships	No	Yes	Yes
Overfitting risk	Low	High	Low
Feature importance	Coefficients	Yes	Yes
Computational cost	Low	Low	High
Handling interactions	Poor	Good	Excellent
Scalability	High	Medium	Medium

in education and decision support [9].

In the context of startups, simulations typically evolve over discrete time steps. At each step, the system updates its state, evaluates conditions, and applies decision rules. The results are then used to generate feedback for the user.

An important feature of such systems is the presence of feedback loops, where past decisions influence future outcomes. This dynamic behavior helps users understand long-term effects and complex interactions.

1.7 Synthesis

This chapter has presented the main theoretical foundations required for the design of a startup failure simulation platform. Startup failure is a complex and multidimensional phenomenon influenced by financial, market, strategic, organizational, and external factors.

A systemic approach is necessary to capture the interactions between these factors, while simulation techniques provide a practical way to model and explore them. Decision-making systems, whether rule-based or data-driven, play a central role in evaluating user actions and generating realistic outcomes.

The three machine learning algorithms examined (logistic regression, decision trees, and random forests) offer complementary strengths for modeling startup failure. Logistic regression provides interpretability and statistical rigor, decision trees offer intuitive visualization of decision rules, and random forests deliver superior predictive accuracy through ensemble learning.

These concepts form the basis for the design and implementation of the proposed platform, which aims to provide both analytical insights and educational value.

Chapter 2

Study of Existing Solutions

2.1 Introduction

In recent years, several tools and platforms have been developed to support entrepreneurship, decision-making, and business training. These systems aim to help users understand complex business environments and improve their strategic thinking. This chapter analyzes existing solutions related to startup simulation, decision support systems, and educational platforms.

The objective of this study is to identify the strengths and limitations of current approaches in order to justify the design choices of the proposed simulation platform.

2.2 Business Simulation Platforms

Business simulation platforms are widely used in education and professional training. They allow users to experiment with business decisions in a virtual environment and observe their consequences.

Examples of such platforms include:

- **SimVenture Evolution**: A business simulation tool that allows users to manage a virtual company, including finance, marketing, and operations.
- **Capsim**: A widely used platform in business schools that simulates competitive markets and strategic decision-making.
- **Marketplace Live**: A simulation environment where users manage companies in a dynamic market scenario.

These systems promote experiential learning and improve decision-making skills [9]. However, they are generally designed for established businesses and do not specifically focus on startup failure scenarios.

2.3 Startup-Oriented Platforms

Several platforms and programs are designed specifically to support startups, providing resources, mentoring, and strategic guidance.

Examples include:

- **Y Combinator Startup School**: An online educational platform providing courses and guidance for startup founders.
- **Seed-DB**: A database tracking startup funding and performance.
- **Crunchbase**: A platform that provides data on startups, investments, and market trends.

Reports such as CB Insights [6] analyze startup failures and identify key risk factors. However, these tools are mostly descriptive and do not provide interactive simulation environments where users can test decisions.

2.4 Decision Support Systems

Decision Support Systems (DSS) assist users in making informed decisions using data analysis and modeling techniques.

Examples include:

- **IBM Decision Optimization**: A system that helps organizations make optimal decisions based on constraints and objectives.
- **Microsoft Power BI**: A data analytics platform that supports decision-making through visualization and reporting.
- **SAP Analytics Cloud**: A business intelligence solution combining analytics and planning.

Traditional DSS often rely on rule-based approaches [12], while more advanced systems integrate machine learning techniques [19]. Financial prediction models, such as those proposed by Altman and Ohlson, are commonly used to estimate failure risk [1, 21].

However, most DSS focus on analysis and prediction rather than interactive simulation.

2.5 Simulation and Artificial Intelligence Approaches

Artificial intelligence has significantly improved the capabilities of simulation systems by enabling adaptive and data-driven decision-making.

Examples include:

- **AnyLogic**: A simulation tool supporting system dynamics, agent-based modeling, and discrete-event simulation.
- **NetLogo**: A platform used for modeling complex systems and agent-based simulations.
- **OpenAI Gym**: A toolkit for developing and testing reinforcement learning algorithms in simulated environments.

Machine learning techniques such as Random Forests [3] and reinforcement learning [29] are widely used to model complex systems and optimize decisions. However, these platforms are often technical and not specifically designed for entrepreneurship education.

2.6 Limitations of Existing Solutions

Despite the diversity of available tools, several limitations can be identified:

- **Lack of specialization:** Most platforms are not specifically designed for startup failure analysis.
- **Limited interactivity:** Many tools do not allow users to explore decision scenarios dynamically.
- **Insufficient feedback:** Users often receive limited explanations about the consequences of their actions.
- **Fragmentation:** Simulation, data analysis, and decision-making tools are often separated.
- **Low pedagogical focus:** Some systems emphasize results rather than learning processes.

2.7 Comparative Analysis

To better understand the differences between existing solutions, Table 2.1 presents a comparative analysis.

Platform	Simulation	Interactivity	AI	Startup Focus
SimVenture	Yes	Medium	Low	No
Capsim	Yes	Medium	Low	No
Y Combinator	No	Low	Low	Yes
Power BI	No	Low	Medium	No
AnyLogic	Yes	High	High	No
Proposed Platform	Yes	High	High	Yes

Table 2.1: Comparison of Existing Platforms

2.8 Conclusion

This chapter has presented an analysis of existing solutions, including business simulators, startup platforms, decision support systems, and AI-based simulation tools. While these systems provide valuable functionalities, they suffer from limitations in terms of interactivity, specialization, and pedagogical feedback.

These observations justify the need for a new platform that integrates simulation, decision-making, and learning in a unified environment specifically focused on startup failure scenarios.

The next chapter presents the design of the proposed system.

Chapter 3

System Design

3.1 Introduction

This chapter presents the design of the proposed startup failure simulation platform. It explains the overall system architecture, how the database is designed, and the different sequence diagrams that show the main interactions and simulation workflows of the platform. The goal of this design is to make the system scalable, modular, and able to communicate efficiently between all its components.

3.2 System Architecture

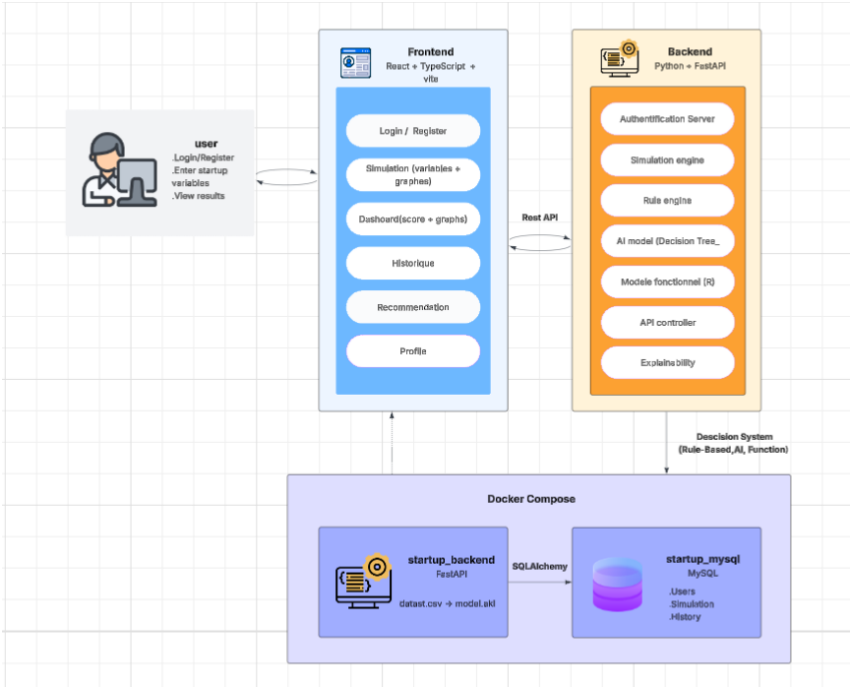


Figure 3.1: Global System Architecture

The proposed platform uses a client-server architecture that is separated into three main layers: the presentation layer, the logic and decision layer, and the database layer. This structure helps make the system more flexible, easier to maintain, and scalable, while also improving communication between the user side and the analytical components.

3.2.1 User and Presentation Layer (Front-end)

The user interacts with the interface built using React and TypeScript, which is the main entry point of the platform. From this interface, users can create accounts, log in, and use the different features offered by the system.

The front-end provides an easy and interactive workspace where startup data, simulation settings, and results are shown in an organized way. It also allows users to navigate smoothly between dashboards, simulation pages, history, and recommendation sections.

In addition, this layer keeps communication between the user and the system running continuously and ensures a responsive and simple user experience. Therefore, the presentation layer plays an important role in improving usability and user interaction with the platform.

3.2.2 Logic and Decision Layer (Back-end)

Implemented using FastAPI and Python, the back-end layer is the main processing and decision-making part of the platform. It manages the business logic of the system and coordinates the different analytical modules used in the simulation process.

At the core of this layer is the simulation engine, which combines several approaches such as a rule-based system, a Decision Tree AI model, and a statistical model. Depending on the selected mode, the system can run one method or compare multiple methods to give a better analysis of startup risk scenarios.

Besides the analysis tasks, this layer also manages authentication, user sessions, and secure access to system resources. It also includes an Explainable AI (XAI) feature that helps explain the results. Instead of only giving scores, it shows why a certain risk level was predicted and which factors influenced the outcome.

So, the back-end layer acts as the intelligent core of the platform by linking data processing, decision support, and system management in one unified structure. .

3.3 Database Design

The database was designed using the Merise methodology to make sure the data is consistent and properly stored. The database layer works as a link between the front-end and the back-end while also ensuring data integrity and consistency.

3.3.1 Conceptual Data Model (CDM)

The Conceptual Data Model (CDM) is a technology-independent view that represents the main entities of the system. The database structure is organized into four main hierarchical entities:

- **User**:handles user login information and authentication details .
- **Startup_profile**: saves startup data and simulation settings.
- **Simulation_run**: represents a single run of a simulation process.
- **Simulation_step**: saves changing simulation values and generated results.

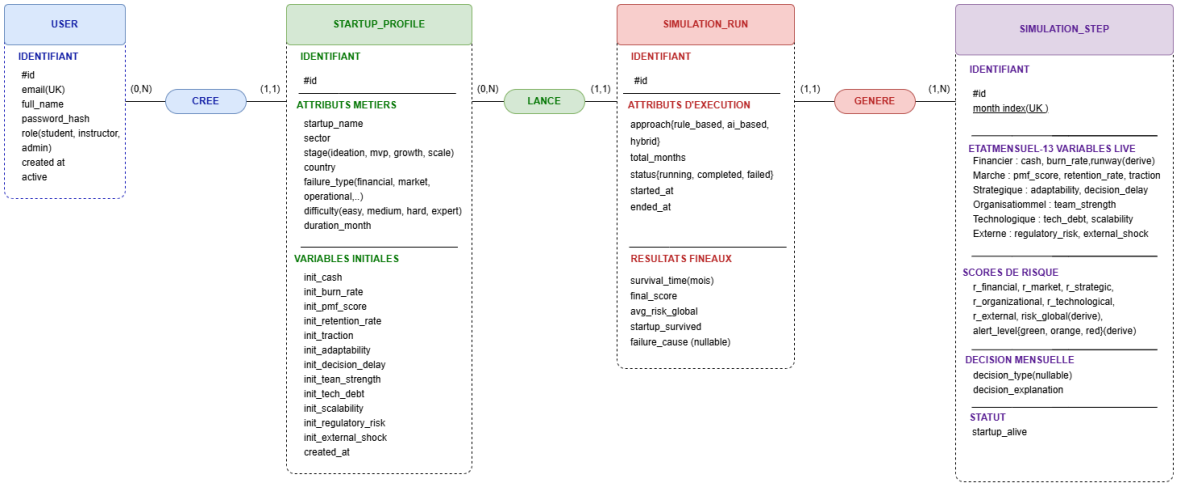


Figure 3.2: Conceptual Data Model (CDM)

The move from the conceptual model to the logical model was done using Merise rules, where primary keys from main entities are transferred into related tables as foreign keys.

3.3.2 Logical Data Model (LDM)

The Logical Data Model (LDM) converts the business rules of the system into a relational database structure optimized for MySQL. It defines how the entities are related, how the data is organized, and how it is stored efficiently for persistent use.

To make it easier to understand and avoid long technical descriptions, the full relational structure—including data types like *INT, VARCHAR, DECIMAL*, as well as Primary Keys (PK) and Foreign Keys (FK)—is shown in the relational schema below.

As a technical optimization method, several additional tracking variables were created such as *runway*, *risk_global*, and *alert_level* are implemented as *GENERATED ALWAYS AS STORED* columns within the database. This design choice reduces repeated calculations during simulation execution and also improves the responsiveness of dashboards and the real-time performance of chart rendering.

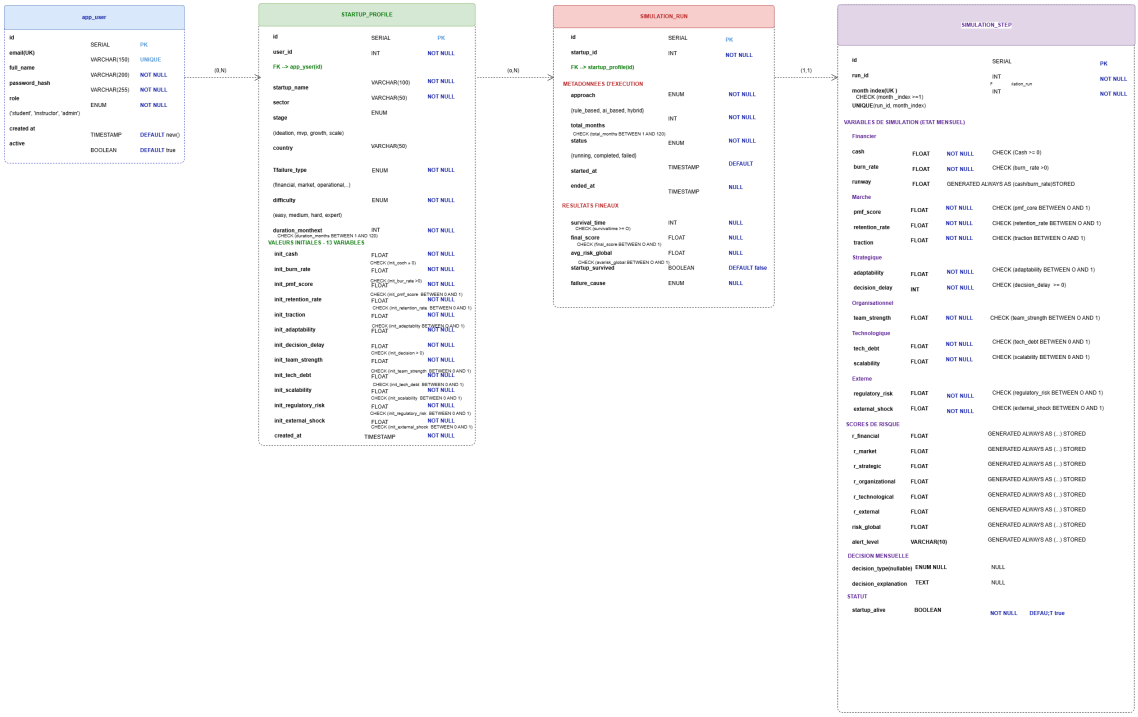


Figure 3.3: Relational Schema of the Logical Data Model (LDM)

3.4 Sequence Diagrams

This section shows the sequence diagrams that explain the main interactions and simulation processes used in the platform.

3.4.1 Registration Diagram

The registration process lets a new user create an account by entering their personal information. The system checks if the provided data is valid and then saves the user details in the database.

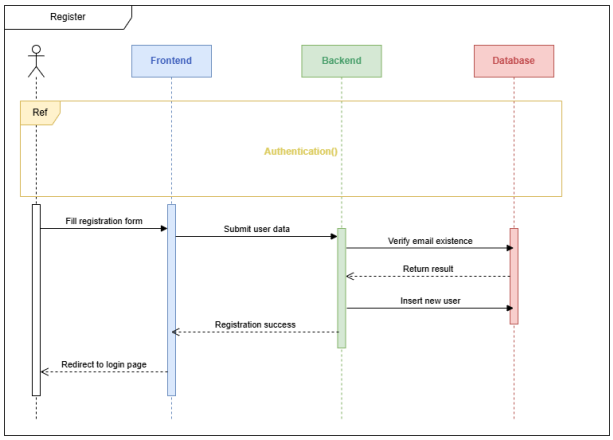


Figure 3.4: Registration Sequence Diagram

3.4.2 Authentication Diagram

The authentication process is used to confirm the user's identity by comparing the entered login details with the information saved in the database. If the credentials match, the user is allowed to access the system.

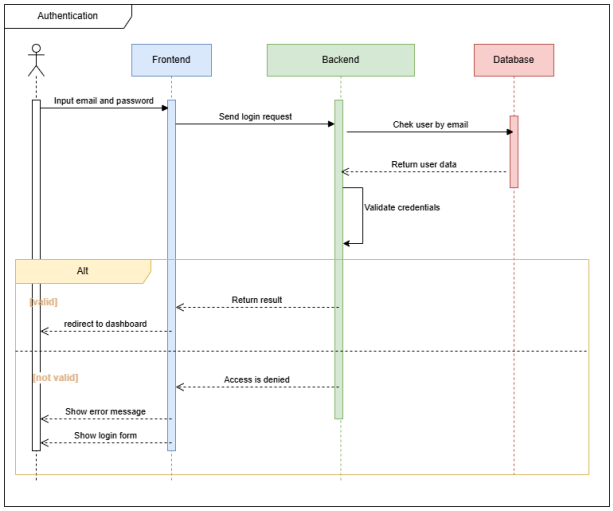


Figure 3.5: Authentication Sequence Diagram

3.4.3 Global Simulation Sequence Diagram

The global sequence diagram shows the full execution process of the startup failure simulation platform by combining the Rule-Based, AI-Based, and Hybrid approaches in one workflow.

The process starts when the user sets the simulation parameters and launches it from the front-end. The request is sent to the back-end, which creates a simulation session and stores the initial data in the database.

During each monthly cycle, the back-end retrieves the previous state and sends the required variables to the simulation engine. Depending on the selected mode, the system

applies rule-based logic, machine learning predictions, or a hybrid approach to update the startup state and generate results.

- **Rule-Based Approach**: The simulation engine uses predefined business rules to update the startup variables and generate results based on fixed conditions.
- **AI-Based Approach**: The simulation engine communicates with the machine learning model to generate prediction results based on trained data and startup variables.
- **Hybrid Approach**: The simulation engine combines predefined business rules with machine learning predictions to produce more accurate and improved simulation results.

After each step, the system stores the generated variables, risk scores, alerts, and decisions in the database. Once all iterations are completed, the back-end computes the final results and sends them to the front-end, where the user can see dashboards, charts, and performance indicators.

Overall, this sequence diagram shows how all system components—front-end, back-end, simulation engine, machine learning services, and database—work together during the simulation process.

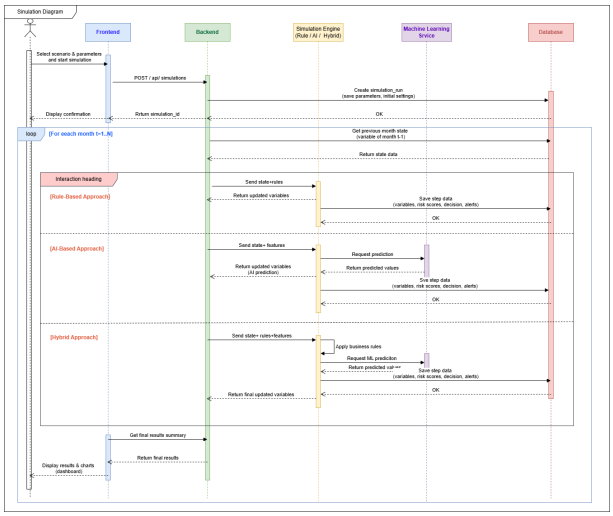


Figure 3.6: Global Simulation Sequence Diagram

3.5 Activity Diagram

This unified activity diagram shows the full workflow of the platform in one view. It starts with user login and authentication, followed by creating a startup profile and setting simulation parameters. After launching the simulation, the back-end initializes the session

and the simulation engine runs iterative monthly steps, updating variables and evaluating risk at each stage.

All data is stored in the database during execution. At the end of the simulation, the system processes the final results and displays them to the user using dashboards, charts, and predictions.

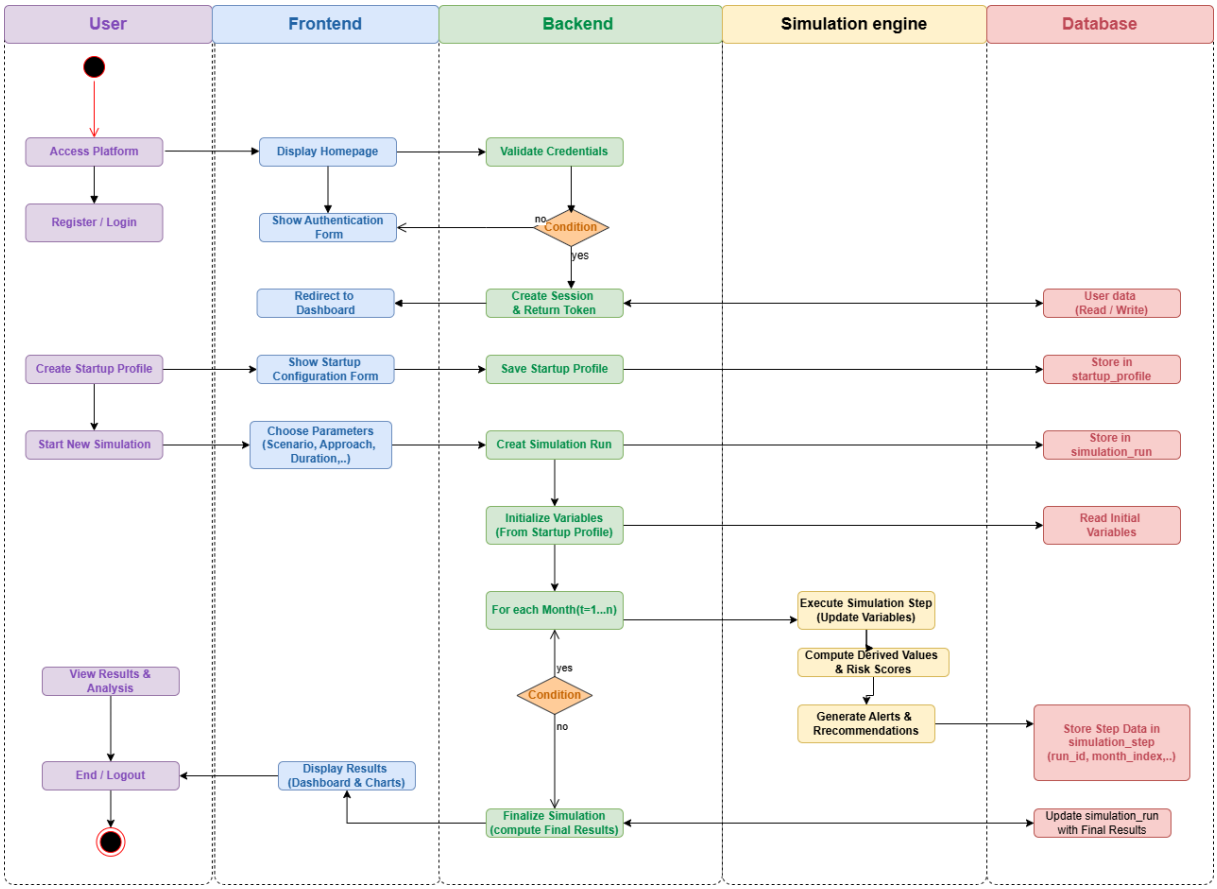


Figure 3.7: Global Activity Diagram of the Startup Failure Simulation Platform

3.6 Conclusion

This chapter explained the design of the proposed startup failure simulation platform. It covered the overall architecture, the database design, and the main sequence diagrams that show how users interact with the system and how the simulation works. The next chapter will focus on the implementation, the technologies used, and the evaluation of the developed platform.

Chapter 4

System Implementation

4.1 Introduction

This chapter explains the implementation of the proposed startup failure simulation platform. It presents the technologies used to develop the system, the implementation of the user interfaces, and how the different simulation modes are integrated. It also introduces the machine learning models used for predicting startup failure and describes the experiments carried out to choose the most suitable prediction approach for the platform.

4.2 Technologies Used

The platform was developed using different modern technologies and tools to ensure good performance, scalability, and easy maintenance.

- **React**: A JavaScript library used for building dynamic and interactive user interfaces [26].
- **CSS**: A style language used to define the presentation and layout of web pages [8].
- **Python**: A versatile programming language used for back-end development and artificial intelligence [10].
- **FastAPI**: A modern Python framework used to build fast and high-performance APIs [23].
- **Docker**: A containerization tool used to deploy applications in isolated environments [16].

4.3 User Interfaces

This section shows the main user interfaces developed for the platform. The interfaces were designed to be simple, interactive, and easy to use, allowing users to run simulations and view startup risk analysis results in an efficient way.

4.3.1 Login Interface

The login interface lets users securely sign in before using the platform's features. It checks the entered credentials and allows access through a secure session if they are correct.

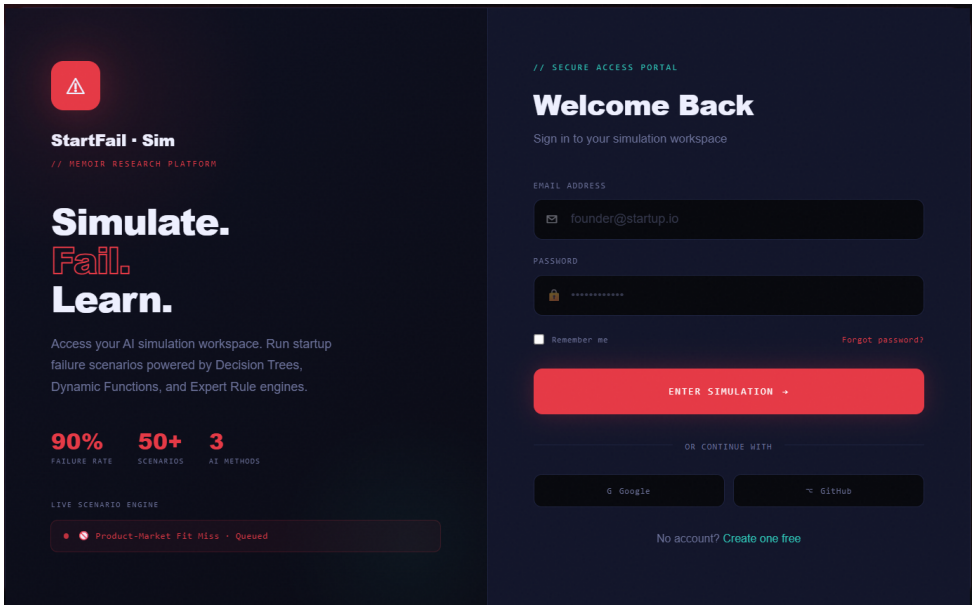


Figure 4.1: Login Interface

4.3.2 Simulation Interface

The simulation interface lets users set startup parameters and choose the simulation mode they want. The platform provides three different execution methods:

- Artificial Intelligence Simulation
- Rule-Based Simulation
- Hybrid Simulation

Users can adjust the startup settings and run simulations dynamically to study how the startup changes over time and how its risk evolves.

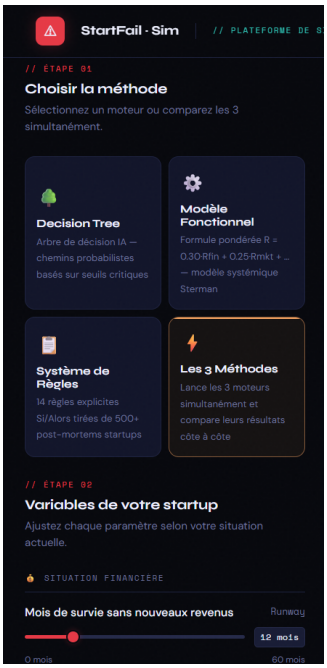


Figure 4.2: methods



Figure 4.3: variables

4.3.3 Dashboard Interface

The dashboard interface gives a visual overview of startup performance indicators, simulation history, alerts, and analytical summaries. It helps users track the evolution of risk factors and understand the results in real time.

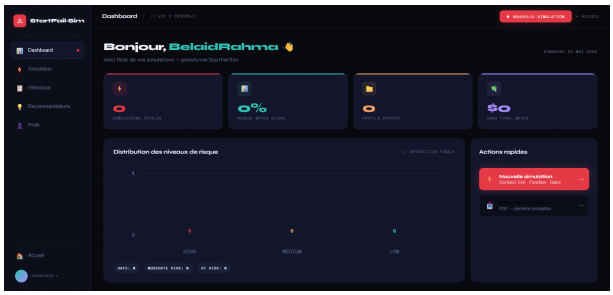


Figure 4.4: Dashboard Interface

4.3.4 AI Prediction Result Interface

The result interface shows the outputs produced by the AI-based simulation model. It includes startup risk predictions, key indicators, and explanations that help users understand how the decisions were generated. .

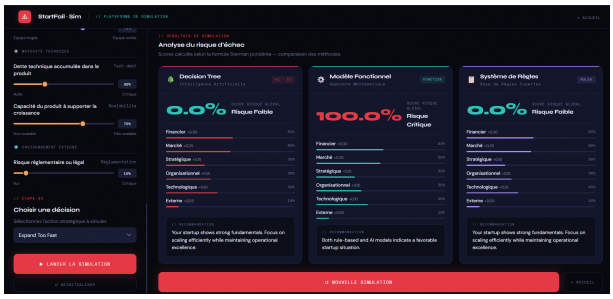


Figure 4.5: AI-Based Simulation Result Interface

4.4 Machine Learning Evaluation

The high failure rate of startups has created a growing need for intelligent systems that help analyze risks and support decision-making in entrepreneurship. In this context, machine learning offers useful methods to predict startup outcomes based on different business and financial factors.

This section compares three supervised machine learning models: Decision Tree, Random Forest, and Logistic Regression, to find the most suitable one for the proposed startup failure prediction platform.

The evaluation considers not only accuracy, but also how easy each model is to interpret and how well it fits within the simulation platform.

4.5 Dataset Overview

The dataset used in this study is the Startup Success Prediction dataset collected from Kaggle. It includes startup-related information that is used to test the performance of the proposed platform and to identify the factors that can affect whether a startup succeeds or fails. The main parts of the dataset are summarized as follows:

- **Data Characteristics**: The dataset contains 13 main features that correspond to financial, market, organizational, and external risk factors defined in the theoretical framework.
- **Target Logic**: The problem is defined as a binary classification task (0: Survived, 1: Failed), while keeping a 62.3% failure rate to represent the high-risk nature of real startup environments.
- **Data Quality and Partitioning**: The dataset does not contain any missing values and it is divided into 80% for training (800 samples) and 20% for testing (200 samples) to ensure a fair evaluation.

- **Predictive Performance Metrics**:Because the classes are imbalanced, the models are evaluated using Accuracy, Precision, Recall, and F1-Score to get a more reliable assessment of failure detection performance.
- **System Operational Metrics**: Deployment suitability is evaluated based on execution efficiency, including training and prediction time (in ms), as well as resource usage, such as CPU consumption (%) and memory usage (in MB).

4.6 Machine Learning Models

4.6.1 Decision Tree

A Decision Tree is a supervised learning method that splits the data step by step based on feature values, creating a tree-like structure made of decision nodes and leaf nodes. Each internal node represents a decision rule, while each leaf node gives the final predicted class. [22]. This model is commonly used in decision-support systems because it is easy to interpret, and it clearly shows how decisions are made when evaluating risk factors. [13].

4.6.2 Random Forest

Random Forest is an ensemble learning technique that builds several decision trees during training using bootstrap sampling and random selection of features. For classification, the final result is decided by combining the outputs of all trees using majority voting. [3]. This method improves prediction performance and reduces overfitting by avoiding reliance on a single decision path, which makes it more suitable for complex risk assessment cases. [13].

4.6.3 Logistic Regression

Logistic Regression is a statistical and machine learning model used for binary classification. It estimates the probability that an instance belongs to a certain class using the logistic (sigmoid) function. It describes the relationship between input variables and the log-odds of the output variable. [14]. Even though it is a linear model, it is still widely used in predictive modeling because it is efficient and easy to interpret. Its coefficients make it possible to clearly understand the effect of each feature. [13].

4.7 Experimental Results

To evaluate the performance of the machine learning models used for startup failure prediction, we used four common classification metrics: Accuracy, Precision, Recall, and F1-Score. These metrics give a complete view of how well the models work by measuring overall correctness and how well each class is predicted, which helps in reliably identifying both failed and surviving startups.

Model	Accuracy	Precision	Recall	F1-Score
Decision Tree	0.955	0.9917	0.9375	0.9639
Random Forest	0.905	0.8978	0.9609	0.9283
Logistic Regression	0.810	0.8358	0.8750	0.8550

Table 4.1: Model Performance Comparison

The Decision Tree achieved the highest accuracy (95.50%), with a very high precision (99.17%) and a strong F1-Score (96.39%). This shows that it can correctly classify startup outcomes with high precision while reducing false positives.

Random Forest achieved a good accuracy of 90.50%, and it obtained the highest recall score (96.09%). This shows that its ensemble method is very effective at detecting most of the actual startup failure cases.

Logistic Regression showed the lowest performance among all the tested metrics, with an accuracy of 81.00% and an F1-Score of 85.50%. This suggests that linear models are less able to capture the complex non-linear relationships and interactions between features in startup failure prediction compared to tree-based models.

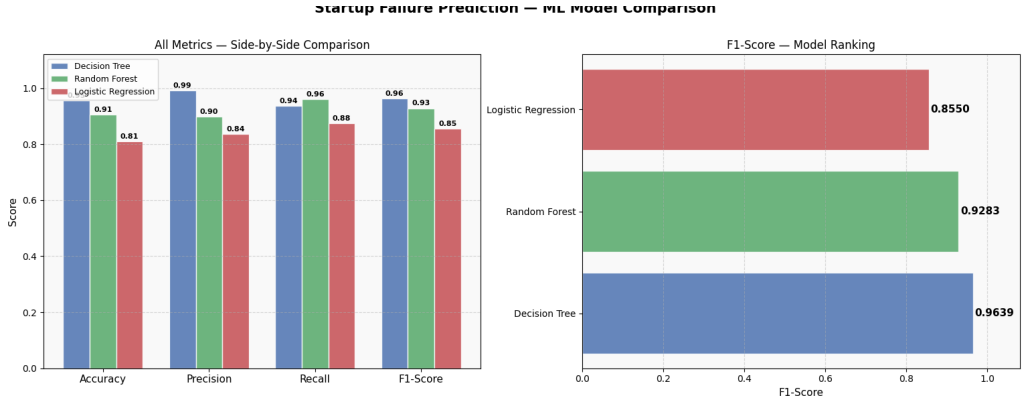


Figure 4.6: ML Model Performance Comparison

The figure shows a visual comparison of the performance of the tested machine learning models using the main classification metrics. It helps to better understand how each model performs in terms of Accuracy, Precision, Recall, and F1-Score, and gives a clearer view of their predictive performance.

4.8 Efficiency Analysis

Besides classification performance, computational efficiency is also an important factor for real-world use, especially in environments with limited resources like processing power, memory, and response time. Therefore, in addition to prediction metrics, each model is also evaluated based on training time, prediction time, CPU usage, and memory usage to determine its suitability for deployment.

Model	Train Time (ms)	Predict Time (ms)	CPU Usage (%)	Memory Used (MB)
Decision Tree	14.921	2.510	1.7	174.59
Random Forest	44.212	30.368	6.3	174.80
Logistic Regression	11.522	0.308	1.4	175.23

Table 4.2: Computational Performance Comparison

Based on the results, the Decision Tree classifier was chosen as the most suitable model for startup failure prediction. It provides the best balance between prediction performance and computational efficiency.

In terms of results, it performed better than the other models, with an accuracy of 95.50% and an F1-score of 96.39%. Its precision (99.17%) and recall (93.75%) are also well balanced, which is important in startup risk analysis to reduce false alarms while still detecting real failure cases.

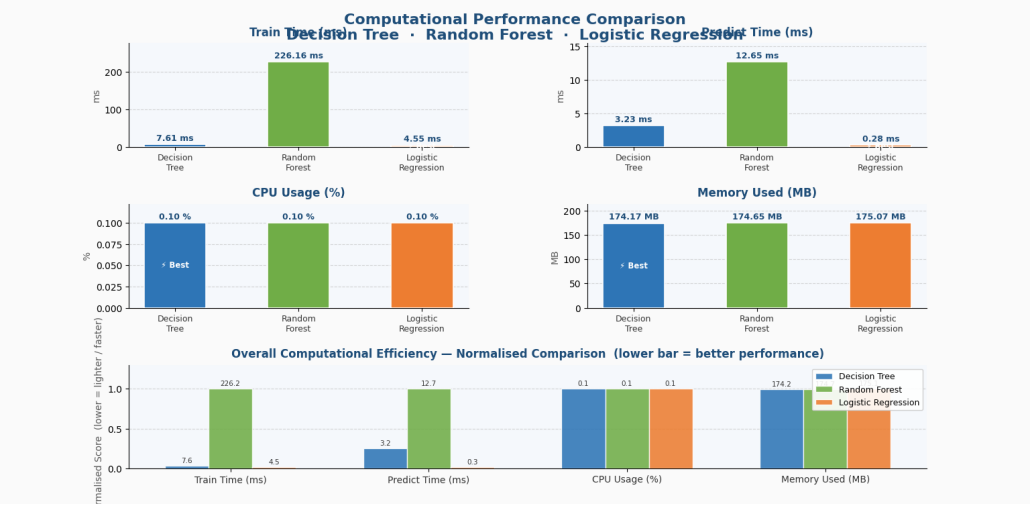


Figure 4.7: Computational Performance Comparison

The figure provides a visual overview of the computational efficiency of the evaluated models. It shows differences in training time, prediction time, CPU usage, and memory consumption, which helps in understanding the trade-offs between performance and resource usage in practice.

4.9 Best Model Selection

Based on the experimental results, the model was selected by comparing both prediction performance and computational efficiency.

The Decision Tree achieved the best overall results with an Accuracy of 95.50%, Precision of 99.17%, and F1-Score of 96.39%, making it the most suitable model for this dataset.

The Random Forest showed a higher Recall of 96.09%, meaning it is better at detecting positive cases, while Logistic Regression had the fastest prediction time.

Finally, the Decision Tree was chosen as the best model for this implementation because it offers a good balance between high performance and easy interpretability.

4.10 conclusion

This chapter explained the implementation of the proposed startup failure simulation platform, including the technologies that were used, the designed user interfaces, and the integration of different simulation methods.

It also presented a comparison between several machine learning models to determine the most suitable model for predicting startup failure.

The experimental results showed that the Decision Tree model achieved the best balance between prediction accuracy, execution efficiency, and ease of interpretation. Therefore, it was selected as the most appropriate model to be integrated into the proposed platform.

General Conclusion

Startups usually operate in very uncertain and changing environments, where financial problems, poor decisions, market shifts, and operational issues can quickly lead to failure. Understanding these factors is important to improve decision-making and to better identify startup risks.

The main goal of this project was to design and build an interactive simulation platform that focuses on startup failure scenarios and risk awareness. To reach this goal, simulation methods, rule-based logic, and artificial intelligence techniques were combined in one scalable system.

First, the theoretical background related to startup failure factors, system dynamics, simulation, and intelligent decision-making was studied. Then, existing solutions were analyzed, and their limits were identified, especially in terms of flexibility, interpretability, and educational usefulness.

Based on this, a client-server architecture was proposed using a React and TypeScript front-end, a FastAPI back-end, and a MySQL database. The platform includes different simulation modes: rule-based simulation, AI-based prediction using machine learning, and a hybrid mode that combines both approaches.

To evaluate the system, different machine learning models such as Decision Tree, Random Forest, and Logistic Regression were tested and compared. The results showed that the Decision Tree performed the best overall, with a good balance between accuracy, efficiency, and interpretability, making it the most suitable model for this platform.

The developed system gives users an interactive way to explore how startups evolve, analyze risk factors, and understand the impact of decisions through simulations and feedback. It also uses Explainable AI (XAI) to make the results clearer and help users understand how predictions are generated.

However, this work still has some limitations and can be improved in the future. Possible improvements include using real startup data, adding more advanced AI methods like reinforcement learning, building recommendation systems, and deploying the platform on the cloud for wider access.

In conclusion, this project shows how simulation and artificial intelligence can be combined to build useful tools that help improve understanding of startup risks and support better decision-making in entrepreneurship.

Bibliography

[1] Edward Altman. "Financial Ratios and the Prediction of Bankruptcy". In: *Journal of Finance* (1968).

[2] Christopher Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[3] Leo Breiman. "Random Forests". In: *Machine Learning* 45 (2001), pp. 5–32.

[4] Leo Breiman et al. *Classification and Regression Trees*. Belmont, CA: Wadsworth, 1984.

[5] CB Insights. *The Top 12 Reasons Startups Fail*. https://www.cbinsights.com/research/startup-failure-reasons-top/. Accessed: 2026-06-26. 2024.

[6] CB Insights. *The Top 20 Reasons Startups Fail*. Report. CB Insights Research Briefs, 2021.

[7] Alex Coad. "Firm Growth and Failure". In: *Small Business Economics* (2014).

[8] MDN Web Docs. *CSS: Cascading Style Sheets*. Consulté le 13/05/2026. 2026. URL: https://developer.mozilla.org/fr/docs/Web/CSS.

[9] A. J. Faria and W. J. Wellington. "Business Simulation Games: Current Usage Levels". In: *Simulation & Gaming* 32.3 (2001), pp. 295–308.

[10] Python Software Foundation. *Python Documentation*. Consulté le 13/05/2026. 2026. URL: https://www.python.org.

[11] Founders Forum Group. *The Ultimate Startup Guide with Statistics (2024–2025)*. https://ff.co/startup-statistics-guide/. 2025.

[12] Joseph Giarratano and Gary Riley. *Expert Systems: Principles and Programming*. Thomson, 2005.

[13] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media, 2009.

[14] David W. Hosmer, Stanley Lemeshow, and Rodney X. Sturdivant. *Applied Logistic Regression*. Wiley, 2013.

[15] Hostinger. *33 Entrepreneurship Statistics in 2026: Global Trends, Challenges & Insights*. https://www.hostinger.com/ng/tutorials/entrepreneurship-statistics. 2026.

[16] Docker Inc. *Docker Documentation*. Consulté le 13/05/2026. 2026. URL: https://www.docker.com.

[17] Maghreb Emergent. *Algérie : le « miracle » des start-up retombe sur terre*. https://maghrebemergent.news/fr/algerie-le-miracle-des-start-up-retombe-sur-terre/. 2025.

[18] Scott Menard. *Applied Logistic Regression Analysis*. 2nd. Thousand Oaks, CA: Sage Publications, 2002.

[19] Tom Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[20] Nils Nilsson. *Principles of Artificial Intelligence*. Tioga, 1980.

[21] James Ohlson. "Financial Ratios and the Probabilistic Prediction of Bankruptcy". In: *Journal of Accounting Research* (1980).

[22] J. R. Quinlan. "Induction of Decision Trees". In: *Machine Learning* 1.1 (1986), pp. 81–106.

[23] Sebastián Ramírez. *FastAPI Documentation*. Consulté le 13/05/2026. 2026. URL: https://fastapi.tiangolo.com.

[24] Eric Ries. *The Lean Startup*. Crown Business, 2011.

[25] Dean Shepherd. "Learning from Business Failure". In: *Academy of Management Review* (2003).

[26] Meta Open Source. *React Documentation*. Consulté le 13/05/2026. 2026. URL: https://react.dev.

[27] Startup Genome. *Global Startup Ecosystem Report*. Report. Startup Genome, 2019.

[28] John Sterman. *Business Dynamics*. McGraw-Hill, 2000.

[29] Richard Sutton and Andrew Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[30] Think Tank CARE. *Les startups et la mue industrielle de l'Algérie*. https://care.dz/fr/espace-presse/les-startups-et-la-mue-industrielle-de-lalgerie. Cited in Maghreb Emergent. 2025.

[31] Unknown. "Failure of Startups in Algeria: An Analytical Study of Internal and External Factors". In: *Afak for Sciences* 11.1 (2026). Available at: https://asjp.cerist.dz/en/article/286790, pp. 65–82.