THE CAPCO INSTITUTE
JOURNAL
OF FINANCIAL TRANSFORMATION

OPERATIONAL

Developers 3.0: Integration of generative AI in software development

FAYSSAL MERIMI | JULIEN KOKOCINSKI

ARTIFICIAL INTELLIGENCE

#58 NOVEMBER 2023

a wipro company

THE CAPCO INSTITUTE

JOURNAL OF FINANCIAL TRANSFORMATION

RECIPIENT OF THE APEX AWARD FOR PUBLICATION EXCELLENCE

CONTENTS

TECHNOLOGICAL

OPERATIONAL

ORGANIZATIONAL

DEAR READER,

As the financial services industry continues to embrace transformation, advanced artificial intelligence models are already being utilized to drive superior customer experience, provide high-speed data analysis that generates meaningful insights, and to improve efficiency and cost-effectiveness.

Generative AI has made a significant early impact on the financial sector, and there is much more to come. The highly regulated nature of our industry, and the importance of data management mean that the huge potential of AI must be harnessed effectively – and safely. Solutions will need to address existing pain points – from knowledge management to software development and regulatory compliance – while also ensuring institutions can experiment and learn from GenAI.

This edition of the Capco Journal of Financial Transformation examines practical applications of AI across our industry, including banking and fintechs, asset management, investment advice, credit rating, software development and financial ecosystems. Contributions to this edition come from engineers, researchers, scientists, and business executives working at the leading edge of AI, as well as the subject matter experts here at Capco, who are developing innovative AI-powered solutions for our clients.

To realize the full benefits of artificial intelligence, business leaders need to have a robust AI governance model in place, that meets the needs of their organizations while mitigating the risks of new technology to trust, accuracy, fairness, inclusivity, and intellectual property. A new generation of software developers who place AI at the heart of their approach is also emerging. Both GenAI governance and these 'Developers 3.0' are examined in this edition.

This year Capco is celebrating its 25th anniversary, and our mission remains as clear today as a quarter century ago: to simplify complexity for our clients, leveraging disruptive thinking to deliver lasting change for our clients and their customers. By showcasing the very best industry expertise, independent thinking and strategic insight, our Journal is our commitment to bold transformation and looking beyond the status quo. I hope you find the latest edition to be timely and informative.

Thank you to all our contributors and readers.

Lance Levy, **Capco CEO**

DEVELOPERS 3.0: INTEGRATION OF GENERATIVE AI IN SOFTWARE DEVELOPMENT

FAYSSAL MERIMI | Managing Principal, Capco
JULIEN KOKOCINSKI | Partner, Capco

ABSTRACT

The concept of "developers 3.0" is emerging, defining the new avant-garde generation of software development professionals. These developers, transcending traditional skills, place generative artificial intelligence (AI) at the heart of their approach, thus revolutionizing software design and development paradigms. This article explores the methodologies and strategies adopted by these innovators, highlighting notable advantages in terms of productivity and quality. At the same time, we address the challenges associated with this combination of traditional software development practices with the new methodologies centered around generative artificial intelligence, such as ethical issues, security concerns, and the need to maintain a balance with traditional skills. Our analysis aims to provide an in-depth perspective on the growing influence of generative AI in the field of software development and its implications for the future of the profession.

1. INTRODUCTION

The history of software development is constantly shaped by revolutionary technological innovations. Today, generative artificial intelligence (AI) stands out as one of the most significant advancements, opening the door to a myriad of questions and possibilities. What are the different types of generative AI and how do they position themselves in the current technological tools landscape? In the face of this rise, does generative AI represent a threat to developers or an opportunity? It is in this context that we introduce the concept of "developers 3.0", innovators who skillfully navigate between traditional skills and advanced AI capabilities. In this article, we will explore the role of generative AI in software development, its benefits, its challenges, and how it shapes the future of the profession.

2. WHAT ARE THE DIFFERENT TYPES OF GENERATIVE AI?

There are two major categories of generative artificial intelligence. The first category encompasses universal generative AIs. These systems are trained on a very large public dataset and aim to address a variety of questions or queries, much like an online search engine. The second category pertains to organization-specific generative AIs. These solutions are designed to access only the data of a specific entity. Typically, these models are also specifically trained to handle particular tasks, such as customer relationship management or the synthesis of legal documents.

In the first scenario, every interaction and feedback from users contribute to the improvement of the underlying algorithm of the generative AI. In the second scenario, interactions only benefit the private version of the organization's own AI, which can be hosted by the provider or by the organization itself.

The market is currently flooded with AI-based tools, but not all are created equal. It is crucial to choose tools that fit one's needs. These tools, spread across various application domains, form a dynamic and constantly evolving landscape, illustrating the depth and diversity of AI-based solutions available to developers. Figure 1 provides an overview of this landscape, highlighting the main categories of tools and how they interact with each other.

Table 1 provides a non-exhaustive list of tools specific to code generation and software development.

Table 1: Tools specific to code generation and software development

CODE GENERATION	
Application	**Description**
aiXcoder	AI-based code completion tool
Bito	Assists developers in code generation
CodeAssist	Offers real-time code suggestions
CodeComplete	Automates code generation for common tasks
CodeGPT	AI model for code generation
Codel	Facilitates code writing with AI-based suggestions
Codeium	Intelligent code completion tool
GitHub Copilot	Coding assistant powered by OpenAI
CodeGuru	Amazon tool for AI-based suggestions
Mutable	Code generator based on project needs
Replit Ghostwriter	Offers code suggestions while writing
Tabnine	AI-based code completion for various IDEs
Warp AI	Speeds up the coding process with automated suggestions
CODE ANALYSIS AND DEVOPS	
Application	**Description**
Adrenaline	Analyzes code to optimize performance
AI Code Reviewer	Examines code for potential errors
Codacy	Code quality analysis platform
Codeball	Analyzes code to improve quality
Coderbuds	Collaborative code analysis tool
Codiga	Checks code quality in real-time
Metabob	Analyzes code to detect bugs
What The Diff	Compares code versions to detect changes
Whispr	AI-based code analysis tool

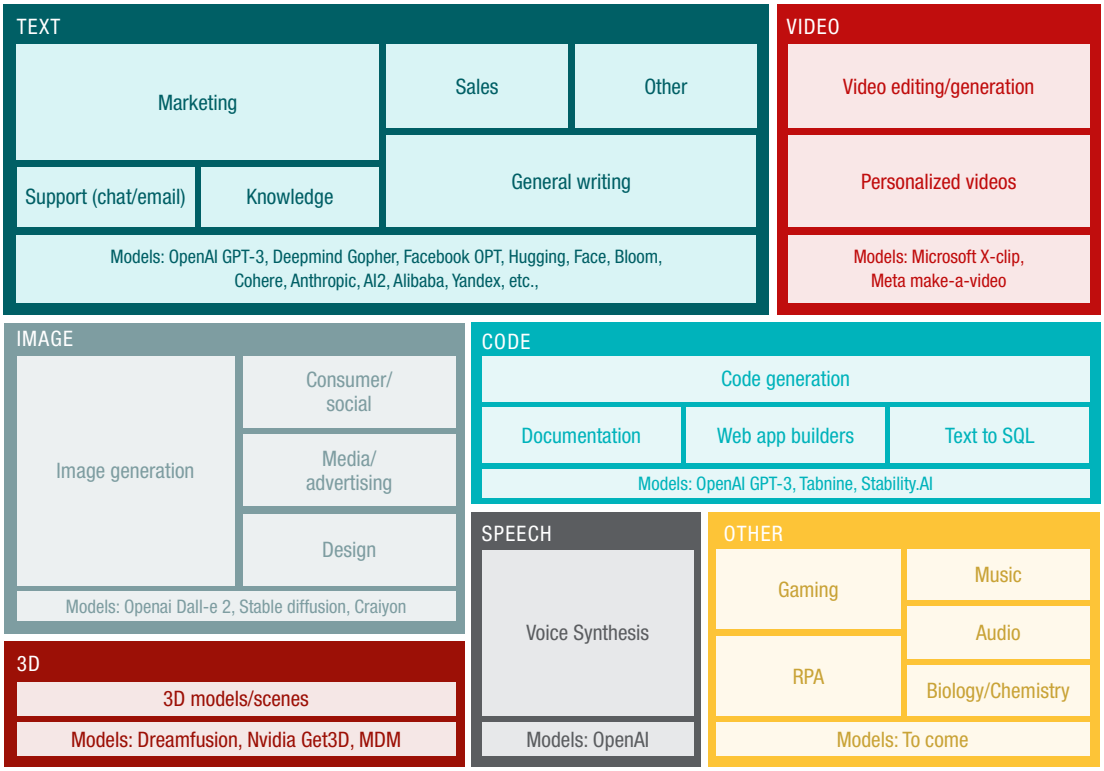
DOCUMENTATION GENERATION	
Application	**Description**
DocumentationLab	Automatically generates documentation from code
DocuWriter	AI-based documentation writing tool
FigStack	Creates technical documents with AI-based suggestions
Mintlify	Transforms code into readable documentation
Stenography	Automated documentation tool for projects
DATA	
Application	**Description**
AI2sql	Converts natural language questions into SQL queries
Channel	Analyzes data to provide insights
Chat2Stats	Transforms conversations into statistics
Consensus	Collaborative data analysis tool
Dataherald	Generates reports based on data analysis
Defog	Clarifies data for better understanding
GenerativeBI	AI-based data analysis tool
Finalle	Automated data analysis platform
Kanaries	Transforms data into visualizations
Lookup	AI-based data search tool
Maya	Assists in data analysis with AI-based suggestions
ProbeAI	Real-time data analysis platform
SQL Genie	Assists in creating SQL queries
SQL Genius	Suggests optimizations for SQL queries
String	Textual data analysis tool
Symbl	Converts conversations into analyzable data
TableTalk	Transforms data into natural language speech
Windsor	Visual data analysis platform

3. IS GENERATIVE AI A THREAT TO DEVELOPERS?

Generative AI, although a powerful and constantly evolving technology, is not necessarily a threat to software developers. In fact, there are reasons to believe that generative AI can be viewed as an asset, rather than a threat, to developers:

- **Complementarity rather than substitution:** generative AI is designed to complement developers' skills, not replace them. It can automate certain repetitive tasks, but human creativity, logic, and expertise remain indispensable for many aspects of software development.
- **Complexity of real projects:** while AI can generate code for specific tasks, the complexity, architecture, and business logic of real projects require deep understanding and human expertise.
- **Validation of AI-generated code:** regardless of the source of the code, whether generated by AI or written manually, rigorous control is essential. Developers must always check, test, and validate the code to ensure its quality and functionality.

Figure 1: The generative AI application landscape



Source: Derived from Sequoia Capital[1]

- **Human interactions:** software development is not just about code. It is also about understanding client needs, working as a team, communicating ideas, and solving problems together. These interpersonal skills cannot be replaced by a machine.
- **Ethical and moral boundaries:** there are decisions in software development that require ethical and moral judgment, such as considering user privacy or creating socially impactful applications. These decisions require human thought.
- **Adaptability:** needs and technologies are constantly evolving. Developers can adapt, learn, and change direction based on changing requirements, a flexibility that AI has not yet achieved.

According to a recent survey conducted by Stack Overflow, which asked the opinions of over 90,000 developers regarding the adoption of generative AI,[2] AI is already firmly rooted in the daily lives of developers. The study finds that 44% of them actively use AI-based tools in their processes, and another quarter plan to follow this trend soon. It is particularly notable that 55% of programming novices turn to these tools from the start of their learning.

However, the issue of trust remains concerning. Only 3% of the developers surveyed have absolute trust in these AI tools for development. Conversely, 6% express total reservations. The majority, or 39%, are cautiously optimistic, falling into a zone of moderate trust.

The study also highlighted regional variations in AI adoption. Developers based in India, Brazil, and Poland are more inclined to integrate these tools, while their counterparts in the European Union and the United States are more reserved.

Looking to the future, the transformative impact of AI on software development is undeniable. An impressive majority, 77% of the developers surveyed, anticipate that AI will redefine their way of writing and debugging code in the near future. These results suggest an imminent shift in the software development landscape, with generative AI as a major protagonist.

[1] https://tinyurl.com/yvwrjpfx
[2] https://tinyurl.com/mtsvfktj

4. WHO ARE "DEVELOPERS 3.0" AND HOW DO THEY APPLY GENERATIVE AI?

Developers 3.0 are not just coders; they are innovators. They understand that AI is not a threat but a tool that can be used to augment their capabilities. They are curious, always ready to learn and adapt, and see AI as a collaborative partner rather than a replacement. For instance, take the case of GitHub Copilot, an AI-based tool that suggests lines of code as developers write. Instead of seeing this as a threat, many developers have embraced it as an assistant that speeds up their coding process. Similarly, companies like DeepMind have used AI to optimize energy consumption in data centers, assisting engineers in identifying more efficient solutions. In the realm of game design, AI is used to generate levels or scenarios, allowing developers to focus on other creative aspects of the game. These examples demonstrate how Developers 3.0 integrate AI into their daily work, not as a rival, but as a valuable collaborator.

4.1 Generative AI in software development

Generative AI in software development is transforming the way developers approach and manage their projects. Here, we present a detailed overview of its capabilities and applications:

- **Code generation:** generative AI can produce code from a natural language description, allowing developers to quickly translate their ideas into functional code.
- **Optimization:** it analyzes and optimizes the code to enhance performance, reduce redundancy, and ensure the software operates optimally.
- **Error detection and correction:** AI predicts and identifies potential errors, offering a proactive form of debugging. It also suggests corrections, reducing debugging time.
- **Recommendations:** AI suggests relevant methods, libraries, or approaches for the project, helping developers stay updated with best practices.
- **Automated testing:** it generates and runs tests based on the code, ensuring comprehensive coverage and software robustness.
- **Documentation:** generative AI can automatically generate relevant documentation based on the code, facilitating project understanding and maintenance.
- **Language transformation:** it is capable of translating code from one programming language to another, easing portability and integration across different platforms.
- **Code explanation:** AI can provide detailed explanations about how the code operates, assisting developers in understanding complex, or automatically generated, code segments.
- **Code completion:** by suggesting real-time code segments or structures, AI aids developers in coding more swiftly and efficiently.
- **Simplification suggestions:** AI analyzes the code to pinpoint segments that can be simplified or refactored, ensuring cleaner and maintainable code.

By integrating these capabilities, generative AI offers developers a powerful suite of tools that not only enhance the quality of work but also expedite the development process.

4.2 Illustration of the use of generative AI in software development

Generative AI, with its advanced capabilities, is revolutionizing the way we approach software development. To concretely illustrate its impact, we refer to a recent article written by our colleagues at Capco,[3] which assessed the reliability of AI tools in the software development process using a real-life case of a financial institution. The primary goal of this study was to demonstrate how generative AI can transform technological operational models throughout the software delivery process.

The study focused on six common activities of the software delivery process, including requirement writing, architecture, design, user experience, code writing, testing, and DevOps. Tools developed by OpenAI were used to generate assets typically created during the delivery phase. These assets were then qualitatively evaluated on three dimensions: product quality, time savings, and resource/cost savings.

The results showed that using generative AI for these activities led to significant time savings, especially for code writing. Even in areas where AI's performance was lower, the products generated by AI were useful for verifying the completeness of human-generated assets.

The article concludes by highlighting the potential benefits of using generative AI in the software delivery process, including time savings, improved quality, and cost reduction. The authors recommend that CIOs adopt a systematic approach to integrating AI into the software delivery process, invest in training the necessary skills, and consider using local versions of AI tools to minimize risks associated with using cloud-based versions.

[3] https://tinyurl.com/2fmd569z

5. BENEFITS AND CHALLENGES OF INTEGRATING AI

5.1.Benefits

It is our opinion, and suggestion in this article, that integrating AI within software development has a number of advantages. Some of these advantages are highlighted below.

- **Increased productivity:**
  - **Task automation:** repetitive tasks, such as generating code for common functions or detecting common errors, can be automated. This frees up time for developers, allowing them to focus on more innovative aspects of development.
  - **Debugging time reduction:** with AI tools that quickly identify errors, the time spent debugging is significantly reduced.
- **Quality enhancement:**
  - **In-depth analysis:** AI can scan thousands of lines of code in seconds, identifying errors or inefficiencies that the human eye might miss.
  - **Real-time optimization:** some AI tools can suggest optimizations in real-time, as code is being written, ensuring optimal performance from the outset.
- **Continuous learning:**
  - **Proactive suggestions:** AI can suggest new methods or techniques based on current industry trends, helping developers stay up to date.
  - **Integrated training:** with tools like GitHub Copilot, developers can receive AI-based suggestions as they code, offering a real-time learning opportunity.
- **Enhanced collaboration:**
  - **AI-assisted code review:** AI tools can assist in code review by suggesting improvements or identifying potential issues, facilitating collaboration among team members.
  - **Project management:** AI can also assist in project management by predicting delivery timelines, identifying bottlenecks, and suggesting resource reallocations.
- **Customization and adaptability:**
  - **Coding style adaptation:** some AI tools can adapt to a developer's specific coding style, offering suggestions that match their personal preferences and practices.
  - **Seamless integration:** AI-based tools are designed to integrate seamlessly into existing development environments, offering benefits without disrupting established workflows.

5.2 Limitations and challenges

As with most things in life, nothing comes without challenges, and generative AI is no exception. Below, we highlight some of the challenges that developers face when trying to integrate AI within their systems:

- **Skill loss:**
  - **Skill atrophy:** if developers rely too heavily on AI for routine tasks, they risk losing practice and mastery of certain fundamental skills.
  - **Less hands-on training:** with AI taking over complex tasks, new developers might miss out on essential learning opportunities.
- **Complexity:**
  - **Obfuscated code:** code generated by AI can sometimes be verbose or structured in a way that is not intuitive for humans.
  - **Tool dependency:** if an AI tool becomes obsolete or is no longer supported, it could pose issues for projects heavily dependent on its generated code.
- **Ethics:**
  - **Accountability:** in case of a failure or error caused by AI-generated code, determining accountability can be tricky. Is it the fault of the tool, the developer who used it, or the organization that adopted it?
  - **Transparency:** decisions made by AI are not always transparent, which can pose ethical challenges, especially in sensitive areas like healthcare or finance.
- **Security:**
  - **Potential vulnerabilities:** automatically generated code might introduce unintentional vulnerabilities, exposing applications to risks.
  - **Dependency on external sources:** if AI relies on external data or libraries to generate code, it could introduce unexpected security risks.
- **Cost:**
  - **Initial investment:** adopting AI tools for development might require a significant initial investment in terms of training, licensing, or integration.

  - **Vendor dependency:** once an organization commits to a specific AI tool, they might become dependent on that vendor, potentially leading to additional long-term costs.
- **Interpretability:**
  - **Lack of justification:** AI might suggest or generate code without providing clear justification, making it challenging for developers to understand or justify certain coding decisions.

6. PERSPECTIVES AND THE FUTURE

As generative AI continues to evolve, developers 3.0 will find themselves at the crossroads between tradition and innovation.

This provides new perspectives on the future of these avant-garde developers, who will need to constantly update their skills, to understand the nuances of generative AI, as well as learn how to collaborate effectively with it. Indeed, generative AI will not just be a tool but a collaborator. Developers 3.0 will work in tandem with AI systems, where AI might suggest solutions, optimize code, or even anticipate the developer's needs.

Continuous learning will become the norm for developers 3.0, with an increased emphasis on understanding AI systems and integrating them into software solutions. For example, generative AI will allow for unprecedented personalization of software solutions. Developers 3.0 will then be able to create applications that adapt in real-time to users' needs, offering highly personalized user experiences at speed.

With generative AI increasingly making autonomous decisions, questions of ethics and responsibility will come to the forefront. Developers 3.0 will need to navigate this complex landscape, determining who is accountable for errors or issues caused by AI-generated code.

Impact for organizations will be massive. On the one hand, there will be challenges as adoption of generative AI increases, requiring new roles, such as "AI collaboration engineer" or "AI ethics specialist", reflecting the changing nature of software development. This will generate a deep cultural change, requiring supporting the organization at all levels. On the other hand, there will be many new opportunities. As an example, by combining crowdsourcing and AI, organizations will pave the way for more advanced and efficient software engineering methods. Leveraging the collective wisdom of a multitude and complementing it with AI skills will bring new benefits, such as accelerated problem solving and higher-quality outputs, and stimulate innovation in software design.

> *"Developers 3.0 integrate AI into their daily work, not as a rival, but as a valuable collaborator."*

7. IMPACT ON THE FINANCIAL SERVICES INDUSTRY

Generative AI is expected to have a significant impact on the financial sector, and developers 3.0 will play a key role in transforming this sector, creating solutions that fully harness the potential of AI.

Financial services, known for stringent regulations and cautious approach to data – be it personal, client, or banking data – demands specialized AI solutions. Most immediate use cases, and easier in terms of compliance and legal validation, would revolve around organization-specific generative AIs. These solutions are designed to access internal data, with the possibility of accessing external data as well, but with the guarantee that no sensitive information is transmitted outside of the institution. Such solutions could address existing pain points while allowing institutions to experiment with, and learn from, generative AI.

For instance, knowledge and document management can greatly benefit from this technology, enabling all departments to swiftly obtain answers based on internal intelligence and experience.[4] Within IT application maintenance teams, IT developers and analysts powered by generative AI will become more effective in solving bugs and issues, leveraging past experience and data.

Regulatory compliance is a constantly evolving challenge in the financial services sector. Generative AI can assist banks in this area as well by generating checklists and updates aligned with the latest regulatory directives. It can also bolster fraud detection by generating scenarios based on known fraud patterns, thereby identifying potential system vulnerabilities, and proposing new solutions, rules, and strategies.

[4] https://tinyurl.com/34zmcdnb

Image generated by Adobe Firefly

Customers interactions will significantly evolve thanks to generative AI. Relationship managers, call centers, claim management teams, or anyone having interactions with customers and collaborating with generative AI will be able to improve customer experience and satisfaction, through quicker, more precise and tailored communication and propositions. Generative AI will also power more sophisticated consumer-facing chatbots.

Moreover, by anticipating market and consumer trends, generative AI can accelerate the creation of innovative financial products tailored to consumers' evolving needs.

Developers, especially developers 3.0, are pivotal in this transformative journey. The future of banking and insurance envisions adaptive systems. These systems, powered by generative AI, can realign in real-time to economic shifts, ensuring efficient asset and liability management.

The advent of universal generative AIs in software development offers financial institutions a unique opportunity to modernize legacy systems cheaper and faster. Developers 3.0 can interpret current code, even in languages that might have become obscure, draft specifications based on this code, hasten adaptations with the users, and transition this code to newer technologies at speed.

With the world's markets becoming increasingly interconnected, developers 3.0 will leverage generative AI to devise highly secured tools and solutions that preemptively manage risks associated with global fluctuations and threats.

In the evolving landscape of the financial services sector, the expertise of developers 3.0 combined with the capabilities of generative AI promises a future marked by efficiency, security, and personalization. As financial institutions steer through the complexities of modernization and regulatory adherence, generative AI emerges as a symbol of innovation and forward momentum.

8. CONCLUSION

The advent of generative AI marks a pivotal milestone in the evolution of software development. The concept of "developers 3.0", which we introduced in this article, epitomizes this new breed of professionals, who, equipped with traditional skills and a profound understanding of AI, are poised to redefine the boundaries of what is achievable in the software development realm.

These trailblazing developers are not merely technicians, they are visionaries who recognize the potential of generative AI as a collaborative tool, rather than a threat. They stand at the nexus of human-machine synergy, working hand-in-hand with AI systems to craft solutions that are more innovative, efficient, and tailored.

However, as we have discussed, this evolution does not come without its challenges. Ethical dilemmas, accountability, and the need for ongoing education and cultural change arise, demanding thoughtful and coordinated efforts from the developer community, businesses, and policymakers.

Ultimately, the future of software development is bright and brimming with uncharted possibilities. The developers 3.0 are at the forefront of this new era, and we eagerly anticipate the innovations and transformations they will bring to industry and beyond.

ABOUT CAPCO

Capco, a Wipro company, is a global technology and management consultancy focused in the financial services industry. Capco operates at the intersection of business and technology by combining innovative thinking with unrivalled industry knowledge to fast-track digital initiatives for banking and payments, capital markets, wealth and asset management, insurance, and the energy sector. Capco's cutting-edge ingenuity is brought to life through its award-winning Be Yourself At Work culture and diverse talent.

To learn more, visit www.capco.com or follow us on Facebook, YouTube, LinkedIn and Instagram.

WORLDWIDE OFFICES

APAC
Bangalore – Electronic City
Bangalore – Sarjapur Road
Bangkok
Chennai
Dubai
Gurgaon
Hong Kong
Hyderabad
Kuala Lumpur
Mumbai
Pune
Singapore

EUROPE
Berlin
Bratislava
Brussels
Dusseldorf
Edinburgh
Frankfurt
Geneva
London
Milan
Munich
Paris
Vienna
Warsaw
Zurich

NORTH AMERICA
Charlotte
Chicago
Dallas
Hartford
Houston
New York
Orlando
Toronto
Washington, DC

SOUTH AMERICA
Alphaville
São Paulo

THE COVER IMAGE WAS CREATED USING JASPER AI, AN AI ART GENERATOR

FSC
www.fsc.org
MIX
Paper from responsible sources
FSC® C004309

WWW.CAPCO.COM

CAPCO 25
a wipro company