packt

UX for Enterprise ChatGPT Solutions

A practical guide to designing enterprise-grade LLMs

RICHARD H. MILLER, PH.D.

Foreword by Jeff Johnson, Ph.D.
Former Professor, Department of Computer Science, University of San Francisco

UX for Enterprise ChatGPT Solutions

A practical guide to designing enterprise-grade LLMs

Richard H. Miller, Ph.D.

‹packt›

UX for Enterprise ChatGPT Solutions

Group Product Manager: Niranjan Naikwadi
Publishing Product Manager: Tejashwini R
Book Project Manager: Aparna Ravikumar Nair
Content Development Editor: Manikandan Kurup
Technical Editor: Kavyashree K S
Copy Editor: Safis Editing
Proofreader: Manikandan Kurup
Indexer: Rekha Nair
Production Designer: Gokul Raj
Senior DevRel Marketing Executive: Vinishka Kalra

First published: September 2024

Production reference: 1220824

Published by Packt Publishing Ltd.
Grosvenor House
11 St Paul's Square
Birmingham
B3 1RB, UK

ISBN 978-1-83546-119-8

www.packtpub.com

It is no surprise to anyone who knows me that I dedicate this book to my loving wife, Jill, and my two amazing kids, Madison and Max. They make me proud every day, even though they don't know what I do for a living.

For those who know about UX, UXD, UI, experience design, or whatever flavor of the month we call our field, I dedicate this book to all the design professionals who go to great lengths to understand their customers and apply exceptional design practices to make their customers successful.

- Richard H. Miller, Ph.D.

Foreword

Anyone who has tried ChatGPT, Google's BARD, or any other **large language model** (**LLM**) knows that getting useful answers from them requires knowing how to feed the LLM the relevant inputs and formulate the right queries (called "*prompts*" in the LLM world). The desire to create LLM-based applications that are actually useful and not just entertaining has given rise to a new field of expertise: conversational design.

Riding the AI wave, AI/LLM experts are churning out books and courses on how to incorporate AI and LLMs into software applications, e.g., chat systems, smart speakers, and business software. However, just knowing how to focus an LLM on a specific domain and how to compose instructions for it is insufficient to create applications that people can easily learn and use productively. That requires a separate type of expertise: how to design applications to meet user and task requirements, often known as **User Experience Design** (**UXD**) or **User/Task Centered Design** (**UCD**).

Of course, there are many training courses and books on UXD and UCD. As the author of some, I can say that perhaps too many. However, UXD and UCD books don't teach how to incorporate LLMs into focused applications.

Richard Miller has extensive experience in both LLMs and UXD/UCD, so his book is unique: it blends those two seemingly disparate disciplines, teaching *both*—how to create and incorporate into enterprise applications specialized versions of ChatGPT that focus on domains relevant to the application, *and* how to ensure that those applications are easy to learn and use, meet the requirements their intended users, and provide value for the enterprises that deploy them.

The book is structured as a tutorial on building ChatGPT-based enterprise applications, interspersed with lessons on the methods used in UXD and UCD. It starts by summarizing the histories of AI/LLMs and UXD/UCD and explains the benefits of each, but then jumps into a tutorial on creating a custom instance of ChatGPT with proprietary data. Subsequent chapters teach how to perform user research and task analysis, prioritize features and improvements, choose the most suitable type of application, how the book's recommendations fit into Agile development processes, and more.

One of the book's most useful features is that it is designed mainly as an e-book with live links to sources, examples, and other external resources. For the benefit of readers of the book's printed version, Richard created a webpage with all the links in the e-book version.

Conclusion: This book is the first of its kind and a significant and welcomed addition to the growing body of books on maximizing the value of LLMs.

- Jeff Johnson, Ph.D.

Former Professor, Department of Computer Science,

University of San Francisco

Acknowledgment

From the day I started this book-writing journey, it has been a wild ride. I appreciate the efforts of the Packt publishing team, who reached out to inquire about me writing this book. It started as a book on how to use ChatGPT to help be a good designer, but the more valuable contribution to our field is how to make ChatGPT do what we want in the enterprise. Thank you to Aparna, Tejashwini, Vandita, Shambhavi, and Manikandan for making this process easy.

The book has a lot of UX specifics, and I certainly don't want to understate the value of a good technical review team. Dan Miller and Martin Yanev brought thoughtful insight to the UX chapters, which were mainly new to them while helping me refine the more technical portions of the book. Kevin Mullet's book, which he authored with Darrell Sano, *Designing Visual Interfaces*, was so thoughtful and insightful that I never thought I could write a book myself. However, his great efforts in his technical review of this book dramatically improved how I wrote and thought about this book from the reader's perspective. Also, I thank Jeff Johnson, a pillar of the UX community, for his extraordinary effort to include his thoughts in the forward for this book. His wisdom has already been so insightful.

The feedback from the Wove.com team, especially Jay Edlin and David Xu, hugely improved our in-depth case study. In my need to reach out and get approvals for images and references for the book, I have to thank a slew of authors and AI experts for allowing me to share some of their work in the book. Thank you to Dan Miller from Opus Research for allowing me to quote his Conversational AI Survey, Chris Spalton for his amazing UX cartoon storyboards, Christian Roher from `xdstrategy.com` for his landscape of user research methods, Mathew Leverone from ScaledAgile for the various Agile material, Jakob Nielsen for being so open with his usability heuristics, Keyvan Mohajer and Fiona McEvoy for the SoundHound image, Ryan Patrick from Occamonics, Haofen Wang for his images from their paper on RAG, Kevin Dewalt from Prolingo for the image from his Lessons Learned video, Chen Qian for the ChatDev image, Jindong Wang for their figure from his article, Jim Ekanem for his insights into accessibility, Mihael Cacic for his wonderful training class and use of his graphics and fine-tuning examples, and Joe Huang for the ODA demo screenshot.

In addition, I have learned so much from my peers: linguists, writers, engineers, developers, designers, researchers, and engineering leaders; there is probably no one idea here that wasn't touched by their expertise. Thank you to Toff van Alphen, Andrew Bulloch, Juliette Fleming, Jason Fox, Miranda Glasbergen, Jason Goecke, Philip Hayne, Joe Huang, Peggy Larson, Jacob Nielsen, David Price, Ken Rehor, Grant Ronald, Dalila Rosales, Aita Salasoo, Ben Schneiderman, David Stowell, Bruce Tognazzi, and Hardeep Walla. I have learned so much from y'all on my journey.

Icons for some images were provided by flaticon.com:

- Speaker icon by Eklip Studio (`https://www.flaticon.com/free-icon/audio-input_13430774?term=voice+input&page=1&position=6&origin=search&related_id=13430774`)
- Edit icons created by Kiranshastry (`https://www.flaticon.com/free-icons/edit`)
- Food icons created by Freepik (`https://www.flaticon.com/free-icons/food`)

Contributors

About the author

Richard H. Miller, Ph.D., is a dynamic leader in user experience and conversational AI. With over 20 years of experience in UX design strategy and 7 years in conversational AI, he has founded and managed four global teams, delivering user-centered design solutions to Fortune 500 organizations. At Oracle Corp., he led a team that developed the Oracle support portal, generating over $15B of in-service support revenue.

Richard was at the forefront of Oracle's conversational AI deployments on Slack, Teams, and the web. He developed the Expense Assistant AI and designed Oracle's first conversational AI platform. After multiple start-ups, some successful and some not so much, he has grown his design expertise across many disciplines, platforms, toolkits, and technologies. Dr. Miller, as he is known in academic circles or when teaching, specializes in global team building, innovative UX design, Agile design, and growing the expertise of the next generations of UI leaders. Richard still gets to apply what he learned from his Ph.D. in UX design and his MBA. He is committed to excellence and innovation in design and conversational AI.

About the reviewers

Kevin Mullet is a software designer and UX innovator whose user-centered experience designs span a wide range of product types. From GUI platforms (OPEN LOOK) to design systems (the Macromedia User Interface, Oracle's Redwood User Experience), to multimedia authoring tools (Macromedia Director, Extreme 3D), from enterprise applications (Icarian Workforce, Edgenuity, My Oracle Support) to consumer apps and applications (eBay, Kijiji, Parker, Show Evidence, and most recently, Node), there aren't many experience design problems he hasn't run up against over three decades of practice. His latest work on AI-powered conversational design applies his unique perspective and "best of both worlds" approach to supercharging the traditional chat experience.

Martin Yanev is a highly accomplished software engineer with expertise in aerospace and medical technology. With over eight years of experience, Martin excels in developing and integrating software solutions for critical domains such as air traffic control and chromatography systems. As a computer science professor at Fitchburg State University, he has empowered over 280,000 students worldwide. Martin's proficiency in frameworks such as Flask, Django, Pytest, and TensorFlow, combined with his mastery of OpenAI APIs, highlights his instructional prowess. He holds dual master's degrees in aerospace systems and software engineering, driving innovation and advancements in software engineering.

Dan Miller is the founder of Opus Research, where he defines conversational commerce by authoring reports regarding automated speech, natural language processing, conversational AI, analytics, and customer experience.

As the Director of the New Electronic Media Program at LINK Resources (IDC) from 1980-1983, he helped define one of the first continuous advisory services in the information industry. He held management positions at Atari, Warner Communications, and Pacific Telesis Group. He also published Telemedia News & Views, a monthly newsletter regarding developments in voice processing and intelligent telephony.

Dan received a BA from Hampshire College and an MBA from Columbia University.

Table of Contents

5

Defining the Desired Experience 121

Part 2: Designing

6

Gathering Data – Content is King 177

7

Prompt Engineering 227

8

Fine-Tuning 261

Part 3: Care and Feeding

9

Guidelines and Heuristics 297

10

Monitoring and Evaluation 347

11

12

Preface

This book combines **User Experience** (**UX**) expertise with ChatGPT and related **Large Language Models** (**LLMs**) to create enterprise-grade applications that can solve real business problems. This is done in a way that almost all of the learnings of the books will continue to apply to the latest LLMs as they evolve and improve. We focus on the integration of LLMs with business solutions. This includes creating customer chatbots for customer service, creating recommender solutions to offer suggestions for sales and service, making purchase choices, solving other business problems in any vertical, or helping create more effective behind-the-scenes solutions that contain little or no UX. We take the science and art of UI design and research methods, techniques, and recommendations to make LLM solutions functional, usable, necessary, and engaging. Tips and expert secrets on applying UX to every stage of the design of LLM solutions at scale are shared. Almost none of this material is on the Internet or shared at vendor sites, so it is a unique resource for the design and design adjacent community.

Who this book is for

This book would appeal to individuals interested in enhancing their knowledge and skills in UI/UX design and looking for a comprehensive guide incorporating the latest technologies to apply UX principles to create enterprise-grade ChatGPT-powered solutions. It is suitable for seasoned designers looking to expand their knowledge, as well as writers, linguists, product managers, and design-savvy engineers who need to know UI/UX design fundamentals as they apply to ChatGPT.

The book follows a design-centered approach to producing ChatGPT-based solutions to solve business or "enterprise" problems. It helps decide and prioritize customer use cases for generative AI, accelerates the value from an LLM, extends the platform to serve customer needs, and explains monitoring and improving the quality of that service. Learning these skills will give you conversational AI design superpowers. An enterprise or business-class ChatGPT-powered solution should focus on providing customers with a unique skill, something more intelligent and focused than they can get from generic generative AI. To imagine and create world-class LLM-powered solutions, this book is for you.

What this book covers

Chapter 1, *Recognizing the Power of Design in ChatGPT*, begins with a brief introduction to the relationship between design and LLMs, including the art and science of UX and the history of LLMs. The various design frameworks for deploying an LLM are discussed, including a chat UI, a hybrid UI that includes chat with graphical user interfaces, recommender UIs that are not interactive, and designs intended to work behind the scenes with backend solutions. There is a small hands-on lab for building a simple model using a no-code playground.

Chapter 2, *Conducting Effective User Research*, provides many tips and tricks for using some of the most critical user research tools to evaluate adding ChatGPT and LLMs to enterprise solutions. We cover methods such as surveys, needs analysis, interviews, and digging into data to create a conversational analysis.

Chapter 3, *Identifying Optimal Use Cases for ChatGPT*, teaches how to identify the breadth of solutions to which an LLM can add value and explains when an LLM is not suited for a use case. We briefly cover classic use case design and then spend time aligning an LLM's capabilities with user goals. We ensure you know ChatGPT's limitations and biases and how to handle inappropriate responses.

Chapter 4, *Scoring Stories*, helps you become an expert in prioritizing user stories. This chapter is also valuable after a product or service goes to customers. You learn to prioritize updates, patches, and bug fixes so the customer gets the most value from the team's efforts. You will be able to balance customer priorities with the cost of development and make rational decisions to help plan and deliver the most value for the least cost. It explains in simple terms how to apply some special Agile tools to prioritize all this work. No road is without some bumps, but we share some complexities so you can navigate this successfully with the entire team.

Chapter 5, *Defining the Desired Experience*, is the final chapter before we get serious about the inner workings of ChatGPT. You will uncover specific considerations, design issues, and solutions for the full range of contexts of use. These include chat experiences, hybrid UIs (a graphical user interface merged with chat intelligence), recommendation UIs, and backend solutions (those without a customer-facing UI). We will address overarching considerations for these desired experiences, ensuring you know how to handle accessibility and internationalization while creating trust and handling security in any of these solutions.

Chapter 6, *Gathering Data – Content Is King*, dives into the complex nature of enterprise data, which is fundamental to creating a ChatGPT solution based on customers' needs. Explore how data sources such as knowledge bases, databases, spreadsheets, and other systems provide a source of truth. This helps connect customers to actions and explains how product people like yourself can contribute at this stage. Hands-on activities and a case study on annotating and cleaning data help explain the key points. We will cover retrieval augmented generation to help bridge the gap between an enterprise's vast data sources and the LLM.

Chapter 7, *Prompt Engineering*, coaches you on creating instructions that control, adapt, and personify the communications from the LLM to the customer. You will learn the difference between prompts anyone can give to an LLM and the more refined nature of instructional prompts for enterprise solutions.

Chapter 8, *Fine-Tuning*, explains what happens within the fine-tuning process, provides a tutorial on how to start fine-tuning, and continues our in-depth case study. You will be shown different methods to apply when training models. This includes a hands-on exercise to fine-tune a very sarcastic chatbot.

Chapter 9, *Guidelines and Heuristics*, steps past the technical nature of ChatGPT design to examine how to interpret ChatGPT style, tone, and voice. Essential guidelines and heuristics adapted and applied to evaluating ChatGPT solutions are reviewed so you can learn how to use design thinking to create

clarity in the output from your LLM solutions. Dozens of examples are provided, along with a case study and example prompts that tie together the suite of heuristics covered in the chapter.

Chapter 10, *Monitoring and Evaluation*, focuses on knowing if the solution is doing well. It covers evaluating successes and failures, defining quality, and judging whether the UX improves. Our approach is one of care and feeding, following the life cycle of learning from the product's users and feeding back any learnings to have it grow and mature. Statistical measures of model performance, user quality metrics, and heuristic evaluation methods are covered, with tips on improving quality.

Chapter 11, *Process*, focuses on adapting traditional Agile and modern development methods to more interactive and customer-driven needs to improve ChatGPT solutions rapidly. We cover practical strategies to integrate a care and feeding approach into traditional Agile or Agile-like development while explaining why you should advocate for a continuous improvement life cycle.

Chapter 12, *Conclusion*, is the final chapter and provides additional suggestions and coaching to wrap up the entire life cycle covered in this book to set you up for success.

To get the most out of this book

We make no assumptions about any existing use of ChatGPT to build business solutions. We expect everyone to use some form of LLM for personal use. We would like some basic familiarity with UI terms and techniques, even if you are not an expert or a UI professional. We talk about creating surveys, carrying out customer interviews, and giving lots of tips and tricks, but assume a basic understanding of these techniques. References are provided to get you up to speed in places where a knowledge roadblock might appear.

The entire book can be followed without coding; we rely on ChatGPT's free playground experience. We dabble in a few other free resources and LLMs. You will need an account to access our GitHub files and ChatGPT. If you code or are more technical, explore some of the more advanced topics and links provided. Although the book and tutorials are focused on ChatGPT, the learnings in the book can apply to any LLM.

For those in the digital version of the book, you can cut and paste examples directly into ChatGPT. However, no programming or code examples are needed, so missing a comma, for instance, will not impact your ability to learn and follow along. We provide files with sample data; you can use those without issue and test and experiment with the latest LLMs.

We have a solution for readers of the physical book who want quick access to the references and resources or those who notice an out-of-date link because web pages and companies come and go. We are maintaining a single-page online reference guide for all links, articles, demos, videos, and books mentioned in the book. Each chapter has a QR code that links to online references listed from every chapter. This means the links will be able to be updated on the reference website as this emerging field grows.

Online references and links: `Book References` (`https://uxdforai.com/references`)

Download the example and files

You can download the example code files for this book from GitHub at `https://github.com/PacktPublishing/UX-for-Enterprise-ChatGPT-Solutions`. If the files are updated, they will be updated in the GitHub repository.

This chapter's links, book recommendations, and GitHub files are posted on the reference page. `Book References` (`https://uxdforai.com/references`)

We also have other code bundles from our rich catalog of books and videos available at `https://github.com/PacktPublishing/`. Check them out!

Conventions used

Several text conventions are used throughout this book.

`Code in text`: Indicates code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles. Here is an example: "We can compare the answer provided by the base model to the same question answered after adding the Full Thesis.pdf file, a draft of almost 200 pages. ."

A block of code or examples to type into ChatGPT is set as follows:

```
You are a helpful assistant named Alli, short for the name of our
bank. Be courteous and professional. Prioritize information in any
files first. Format output using lists when appropriate.
```

When we wish to draw your attention to a particular part of an example, the relevant lines or items are set in bold:

```
(Mac operating systems only)</li></ul><p><strong>Note:</strong> Our
latest site features will not work with older unsupported browsers. </
p><p>
```

We sometimes include conversations between a user and a chat solution. You can read along by following the standard chat convention: messages sent to the chat are right-justified, and the responses are left-justified.

```
                                  Is solar power a renewable resource?
Solar power is a renewable resource.
Because solar power is an infinite
resource, it has unlimited potential.
```

Bold: Indicates a new term, an important word, or words you see onscreen. For instance, words in menus or dialog boxes appear in **bold**. An example is "Alistair Cockburn's **Writing Effective Use Cases** is the definitive guide when I teach use case design"

Tips, secondary resources, or important notes
Appears like this.

Get in touch

Feedback from our readers is always welcome.

General feedback: For questions about any aspect of this book, email us at `customercare@packtpub.com` and mention the book title in the subject of the message.

Errata: Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you have found an error in this book, we would be grateful if you would report this to us. Please visit `www.packtpub.com/support/errata` and fill out the form.

Piracy: If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please get in touch with us at `copyright@packt.com` with a link to the material.

If you are interested in becoming an author: If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, please visit `authors.packtpub.com`.

Share Your Thoughts

Once you've read *UX for Enterprise ChatGPT Solutions*, we'd love to hear your thoughts! Please click here to go straight to the Amazon review page for this book and share your feedback.

Your review is important to us and the tech community and will help us make sure we're delivering excellent quality content.

Download a free PDF copy of this book

Thanks for purchasing this book!

Do you like to read on the go but are unable to carry your print books everywhere?

Is your eBook purchase not compatible with the device of your choice?

Don't worry, now with every Packt book you get a DRM-free PDF version of that book at no cost.

Read anywhere, any place, on any device. Search, copy, and paste code from your favorite technical books directly into your application.

The perks don't stop there, you can get exclusive access to discounts, newsletters, and great free content in your inbox daily

Follow these simple steps to get the benefits:

1. Scan the QR code or visit the link below

`https://packt.link/free-ebook/978-1-83546-119-8`

2. Submit your proof of purchase
3. That's it! We'll send your free PDF and other benefits to your email directly

Part 1: UX Foundation for Enterprise ChatGPT

Every good story has a beginning, a middle, and an end. In this part, we'll start our journey by exploring how traditional user experience methods and best practices can be applied to creating world-class solutions powered by ChatGPT. We will then explore essential user research methods and provide tips and secrets of the trade that work when creating conversational designs. This will lead us to explore how to define and pick the use cases that **large language models** (**LLMs**) are best suited to solve. A user experience approach is taught to prioritize use cases. When combined with a method to include development costs into the mix, this powerful method from Agile allows for prioritizing the most valuable solutions to build. Although we'll consider how these use cases play out with ChatGPT, almost all the learnings can apply to any LLM model. We'll look at LLM-powered applications, chat-only experiences, and robust chat-powered graphical user interfaces, and we'll even explain how to work with ChatGPT when there is no UI.

This part includes the following chapters:

- *Chapter 1*, *Recognizing the Power of Design in ChatGPT*
- *Chapter 2*, *Conducting Effective User Research*
- *Chapter 3*, *Identifying Optimal Use Cases for ChatGPT*
- *Chapter 4*, *Scoring Stories*
- *Chapter 5*, *Defining the Desired Experience*

1

Recognizing the Power of Design in ChatGPT

If you only like to play with ChatGPT, this book is not for you. Read on to make a quality user experience for customers by incorporating ChatGPT or various alternative language models that span the gamut of cost, quality, and expertise. Every new technology has too many people jumping on the bandwagon only to abandon it with failure. It is widespread. Why? Because they don't know what they don't know. But we know what it takes to make successful ChatGPT solutions for the enterprise. We will take you from zero to hero in your organization. Do you want poor-quality interactions or targeted, intelligent results that resonate with customers? Should they feel empowered and able to explore further with the confidence that they are understood? If it is the latter, this book will focus on user experience design methods, practices, and tools to help decide what to do, design the most effective solutions, and verify that they do what they should do. We can apply user interface practices to the ChatGPT life cycle, leaving you confident to create quality solutions.

This book is intended for designers or design-related professionals, such as product managers, product owners, writers, linguists, or developers, who want to understand how to apply design principles and practices to improve the generative AI experience of customers and employees. For those exposed to design methodologies, they won't be novel, but their application will be. For those with limited exposure to the science of **user experience** (**UX**) design (or UXD), we will provide enough learning to make you dangerous and help create enterprise-grade ChatGPT solutions.

You might have yet to work with generative AI products such as ChatGPT in a production way, maybe only using some of these tools at home or to supplement work. We will only get into some of the explanations of how ChatGPT works in *Chapter 6*, *Gathering Data – Content is King* as we have some ground to cover first. The book follows a typical design. First, we help figure out what to do, prioritize that work, then how to do it, and finally, how to interpret and improve on what was done. We have found the design skills and tips in this book to work for a wide range of design challenges, and through our experience in the last seven years with AI solutions and 30 years of UXD, we have adapted those insights to the creation of generative AI solutions. Following design processes will help one craft high-quality solutions driven by ChatGPT.

In this chapter, we're going to cover the following main topics:

- Traversing the history of conversational AI
- Appreciating the importance of UX design
- Understanding the science and art of UX design
- Setting up a customized model

Technical requirements

There are two ways to work with this book: follow along and learn the principles and practices and use one of the OpenAI playgrounds, which is typically a *no-code* approach, or use the APIs provided by ChatGPT.

We have examples in our book's GitHub repository. If this is your first time using GitHub, it is a place where we will store any materials needed to download to complete the examples in the book. It is an online folder of resources.

GitHub: `Repository for book materials (https://github.com/PacktPublishing/UX-for-Enterprise-ChatGPT-Solutions/)`

GitHub is the repository for all files from the book. Click the download button, highlighted in *Figure 1.1*, to download the file to the desktop. There is no viewer for most of the files on the GitHub repository.

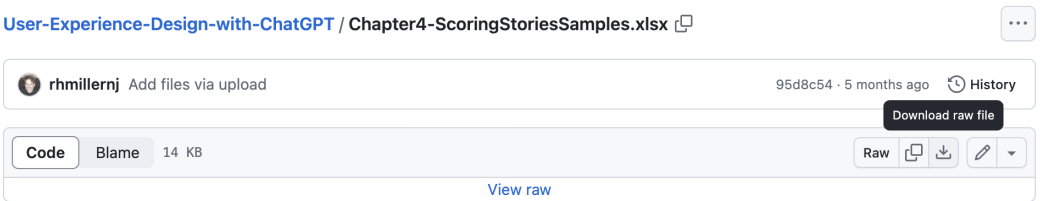


Figure 1.1 – How to download a file from GitHub

Make sure you have a ChatGPT account.

Website: `OpenAI Chat (https://chat.openai.com/)`

That is easy; everyone should have that. We will try out some of the material as we go. This will also allow us to use the Playground, essential for some demos. We have a QR code at the end of each chapter, so all of the references we provide, such as the preceding links, are available online for easier access.

Approach 1 – The no-code approach

You can learn about 80% of this material by just reading, but some folks do better by doing. If you go this way, focus on the design practices and methods and learn how to apply those to any generative AI solution. We will give demos and samples to try without coding. Give the examples a try; understanding how an LLM reacts is critical.

Approach 2 – code with Node.JS, Python, or curl

If you don't already have a ChatGPT account, set one up. Then, head to the quick start guide to ensure Node.js (curl or Python) works. The URL has step-by-step instructions for getting your environment up and running.

Website: `Quickstart guide for developers (https://platform.openai.com/docs/quickstart?context=node)`

> **Note**
> This book does not require coding. A more technical reader can mirror some of our no-code approaches with a code version, but we won't discuss this path.

To use the APIs, follow the instructions on the link:

1. Install the essential software (Node.js, Python, or curl are all documented on the same page; choose the tab that suits you).
2. Install the OpenAI library package.
3. Set up your API key.

And give it a try! If you have used other **Large Language Models** (**LLMs**) or have never used one, try it out and ask anything; we call this input from the user a prompt. *The material in this book can be quickly learned without doing any coding*. Some models don't have the most recent data, so asking about today's weather or sports scores won't work, but if asked to give five ways to clean a clogged toilet, it has the answer. The power we want to expose here is the combination of this powerful experience and UX design practices to create a high-quality, customer-centric experience. Now we have something to discuss!

We should be on the same page concerning the basic history of conversational AI. With all the news, ChatGPT should be well known so we can cover just the basics for a few minutes.

Traversing the history of conversational AI

Interaction design intersected with AI well before the LLM revolution. This history is helpful to appreciate when applying design principles to the latest conversational experiences. Any discussion

of AI at least mentions Alan Turing and the question posed by his article in Mind (a peer-reviewed academic journal).

Article: `Computing Machinery and Intelligence (1950)` (`https://redirect.cs.umbc.edu/courses/471/papers/turing.pdf`)

This is routinely referenced as the **Turing Test**. The ability of a machine to seem intelligent and be indistinguishable from a human.

Article: `Wikipedia on the Turing Test` (`https://en.wikipedia.org/wiki/Turing_test`)

When this was published in 1950, we were still far from a computer being indistinguishable from a human, at least in a text-only interaction. We must skip ahead to the mid-1960s before we see something that appears to engage in discourse.

If we try out one of ELIZA's conversational interfaces from 1964 to 1967, we quickly see its limitations based on its *natural* responses when recognizing keywords or phrases.

Article: `Wikipedia on ELIZA` (`https://en.wikipedia.org/wiki/ELIZA`)

The well-known version is called DOCTOR. It turns written questions asked of it back onto the patient. Give it a try to interact with the psychotherapist chatbot.

Demo: `ELIZA` – The psychotherapist chatbot (`https://web.njit.edu/~ronkowit/eliza.html`)

ELIZA was considered one of the first attempts at passing the Turing Test. With its simple psychotherapist banter to simulate a doctor ("Why do you feel this way?"), it was perceived as human-like. Without going too deep, the discussion around its design looked at the rank of essential words, and it included *transformation* rules that dictated how it treats what the user types. Maybe LLMs are paying homage to this since they are based on **transformers**. We will explain transformers and the terms common to LLMs in later chapters. ELIZA had the superficial appearance of a conversation and could not go *off-topic* or even provide an answer. The psychology of conversational interaction was fundamental to this experience. But it wasn't going to solve anyone's psychological problems. However, things did get better as chatbots; it just took a few decades. Visit Wikipedia to learn a brief history of chatbots.

Article: `Wikipedia's history of chatbots` (`https://en.wikipedia.org/wiki/Chatbot`)

The idea of a *natural language* experience was not lost on the research community from the 1960s to the 2000s. Still, the next step in evolution came with the conversational assistants or chatbots we have seen since around 2016. And this is where interaction design had a significant impact, even though most chatbots were not worth anyone's time. About 100,000 chatbots were created on Facebook Messenger in the first year of its support. I would suggest that 99% of them failed quickly. Very few survived for all the reasons we will explain shortly. But a few lived on when teams were willing to mature the solution. Support use cases, such as for airlines ("How much is a second bag going to cost?" or "Can

I get a refund for a canceled ticket?"), are a great place to answer specific questions with specific answers. Although it seems evident that this can save a company a lot of money in support costs versus a phone call, there is also value to the consumer. For them, time is also valuable. If a customer gets a reliable answer in seconds, they will gladly trade that for 10 minutes of holding onto the phone. It is a win-win. Additionally, this experience can be a frontend for required interactions with humans. In support cases, it can gather details reliably before engaging a human, making it more likely to connect with the right human and give them the details they need to help more directly.

Imagine a young child before going to school. If no one interacts with kids, teaches them, or plays with them, by the time they go to first grade, they lack primary language and interpersonal skills and might not be potty trained. Even the great comedian Steve Martin understood this. Please take a minute and laugh at his bit.

Video: `Steve Martin teaching a kid for the first day of school (https://www.youtube.com/watch?v=40K6rApRnhQ)`

However, remarkable changes can be made by investing in a child's growth and care and feeding them physically and mentally. This maturity is what we can see in chatbots. They won't typically become a Ph.D., but they can coached to be smarter than a 5th grader. We can use design skills to make a chatbot (or any LLM-based solution) knowledgeable, dependable, and articulate. We will apply what we have learned to our next generation of conversational assistants built with ChatGPT. We will critically explore ChatGPT to form robust solutions, and you will learn to notice when there might be other tools out there to use in conjunction with ChatGPT.

There is one other related area worth mentioning. Everyone has experience with phone trees when calling a business. We mentioned this example earlier. Eventually, those "Press 1 for service, press 2 for sales…" gave way to experiences that listened for more than the touch tone of a key press. But how many of us have struggled with these? Probably all of us. Why? Because the experience isn't designed well, and the technology is likely lacking. These, too, will benefit from ChatGPT. So, if you come from creating voice experiences (probably using Voice XML, the de facto standard for modeling interactions for years) or from chatbots within Alexa, Siri, Google, or dozens of other vendors, the learnings and practices of making great experiences apply to ChatGPT. We will go through that extensively in this book.

Wikipedia: `Wikipedia's Voice XML background (https://en.wikipedia.org/wiki/VoiceXML)`

And yet so many of these chatbots, phone trees, or conversational experiences fail to help a primary user accomplish their task. Why? Because of a few critical reasons:

- The features or services in the chatbots don't match the user's needs
- The models don't support the complexity of the user's language
- The user's primary spoken language might not be supported, requiring them to be understood in a secondary language or not be understood at all

- The chatbots only know what they know, so they will return seemingly random results, which is discouraging
- The chatbots do not respond in a voice or tone the customer expects
- The chatbot should have been monitored and improved to address these issues

Your goal should be to set a higher quality bar than expected from a human performing the same task. Sounds crazy? It isn't. A typical support person might be able to help with a narrow topic (say, a website password reset). Another agent would be needed to resolve billing issues or find missing payments. So, the average support person will be less helpful than the future state of well-trained ChatGPT advisors with access to all the institutional data and processes.

This brings us to the founding of companies such as OpenAI. This long history of machine learning models and the increased computational capabilities allow this very large language model to work. OpenAI didn't come into the world view with the December 2022 release of ChatGPT 3.5, the company was founded seven years earlier with non-profit roots. It took over three years to go from GPT-2, which could generate human-like text, to the 3.5 version that gained worldwide attention. For those who like tech history, dive into a brief background on OpenAI.

Article: `The origins of OpenAI (https://www.britannica.com/money/OpenAI)`

Like many Silicon Valley companies, engineers from Google Brain (and DeepMind, which merged with Google), Facebook, and other AI came together at OpenAI. Then, 11 OpenAI employees left to form Anthropic around the start of 2021. None of this happened overnight, so we need to remind ourselves that it will take years for this new technology to weave its way into our everyday lives. The phone, the car, the computer, and the mobile phone have all become fundamental to today's modern society. This will impact all of us more than all these previous inventions, but it will take time to happen. There will be a lot of failures along the way.

Imagine a solution that is 60% accurate at every interaction. Does it sound high for a computer to get something right 60% of the time? Before ChatGPT, some didn't consider this too bad. I routinely used to ask this in my classes (typically from 20 to 100 people per class) on conversational AI. And many folks consider a 50–80% success rate to be "pretty good."

With some simple assumptions, we can understand why these systems fail. Every time a question is asked, the likelihood of a failure increases, as modeled in *Figure 1.2*. To keep this simple, we base this on the independent probability of each turn having the same chance of failure. The system doesn't know it has failed, and if the user trusts it, they might not even notice the failure, thus causing more failures.

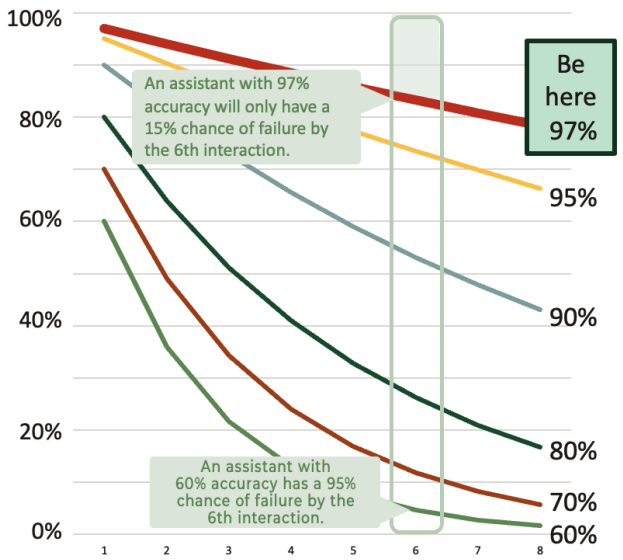


Figure 1.2 – The chance of failure increases at each turn

When asked six questions at a 60% likelihood of success rate, there is a 95% chance of one of those answers being wrong. And what if your next question is dependent on the previous answer? The interaction will go off the rails. I have seen this time and time again. If the user trusts the (wrong) answer, they make the next decision based on that (incorrect) answer. And failure can be assured. This relationship will sour. Customers will go elsewhere if they see these failures (likely a more expensive channel) to get their needs addressed, or worse, they will go to another vendor.

We can consider strategies to improve these curves so that each turn is more likely to succeed. *Chapters 6*, *7*, and *8* explain strategies to use multiple generative AI components to do different forms of validation. Yes, the AI can watch over another AI. While traditional LLMs such as ChatGPT have improved and will continue to improve, we want to provide tools and measurement skills to help ensure success. But look at *Figure 1.2* again. Look at raising the bar to 97% accuracy. After the same number of turns, there is a very good chance (85%) that all interactions were successful. So, raise the bar on expectations.

It is possible to achieve these levels of quality. We will also show how to measure and scope improvements to give the most significant return on investment.

Chatbot failures

To get a laugh at how bad *bad* can be, read this article on chatbot fails. We aim to teach enough design methods to never fall into these black holes of disgrace.

Article: `Chatbot failures (https://research.aimultiple.com/chatbot-fail/)`

With purpose-built experiences, for example, focused on filing business expenses or getting answers to common questions around internal business processes, users spend less energy trying to break the chat experience or ask off-the-wall questions. This behavior, expected in widely available public ChatGPT and chatbot experiences, is likely seen less than 1% of the time when building a custom ChatGPT tool. It will still get questions it might not be able to answer, but it is more likely that they are questions it should *eventually* answer. We will show how to prioritize that backlog to be in the business of continuous improvement.

This brings us to ChatGPT and the new class of LLMs, which are indistinguishable from humans in many ways. Google's LaMDA, Meta's Llama, Anthropic's Claude, and OpenAI's GPT models are all in the same class of software.

- Article: `Wikipedia survey of LLMs (https://en.wikipedia.org/wiki/Large_language_model)`
- Article: `Google's LaMDA (https://en.wikipedia.org/wiki/LaMDA)`
- Article: `Meta's Llama (https://llama.meta.com)`
- Article: `Anthropic's Claude (https://www.anthropic.com/claude)`
- Article: `OpenAI's GPT Models (https://platform.openai.com/docs/models)`

But even if they are like humans, we must ask which humans in the enterprise space they mimic. Does this represent my company? Does it have the knowledge it needs to solve my customers' problems? How will my customer handle a wrong answer? LLMs have a lot of potential and will evolve rapidly. We aim to give you the tools to evaluate whether an LLM solution will fit at every stage of your development.

The importance of UX design for ChatGPT

Does ChatGPT even need an introduction at this point? The innovative model developed by OpenAI is in a new class of LLMs, which are trained on billions of data elements from the internet's vast supply of articles, books, and knowledge. It achieved over 100 million users in about two months. It can generate human-like conversational interactions in text or voice in many languages and converse on vast information. And it does it pretty darn fast.

ChatGPT has undoubtedly come on like a firestorm. Unique, fun, fast, intelligent? However, when designing solutions for your business or enterprise, they should be accurate, have the most current business-related information, and communicate to customers in the voice, style, and tone expected from the business. So, how do you take such a fast-paced moving target and wrap it into a product that exceeds customer's expectations? Can one ensure that it doesn't give random answers that are off-brand? You can, but it takes design. It needs to be monitored. And it would be best if a process was in place to improve it. For that, this is the right place.

Let me define *design* because I see a lot of really horrible definitions. Design is the process and practice of clearly communicating an experience for a user. Good software UX design accounts for human behavior and limitations by applying the scientific method to solving human and machine interface issues. This means we can use what we know about the visual, auditory, and kinesthetic systems and combine them with understanding how the mind works to make decisions on how to create an experience that is functional, usable, needed, and even engaging. We see design all around us: visual design, graphic design, software design, conversational design, building architecture, and many other fields. We use the expertise of user researchers to guide our designs based on subjective and objective feedback from our customers, using formal and informal methods to better understand our users' needs. We then combine the inputs from customers, primary research, and the goals of the product and company and mix in a bit of magic to make great experiences.

If you just put icons on screens or write conversational copy without these efforts, you do production work, not design. We want everything done for a reason. The more done by creating fitness to purpose, the more our customer experience will improve. We don't always get it right. We will get a higher quality product if we know how to fulfill the user's needs. That is where the iterative design concept plays a role. We learn from and improve our designs even if we don't get it right.

UX design, interface design, human factors, user research, **human-computer interaction** (**HCI**), or any flavor of the art and science of interaction design is a collection of experts and expertise that can help shape this functional, engaging, usable, and fun experience. We can build successful chat-based solutions by directly applying the wealth of learnings from these disciplines to "chat" experiences or adapting what we have learned with conversational AI and graphical user experience design to fit into this new world. Using words to communicate with a computer is not new; it has just improved.

This is where you learn how to design a ChatGPT solution for customers based on company knowledge and business needs. Enterprise ChatGPT covers a wide range of experiences. One can be making a support site, a virtual assistant to help employees or job seekers, a sales engagement service that personalizes emails for sales calls, a training application, a product finder or recommender, a tool to analyze legal documents for inconsistencies, or an expert witness tool for lawyers. Code review (evaluating software written in Java or dozens of other popular languages and identifying issues or bugs) is another popular topic in tech. This book will stay away from that use case to focus on more common experiences that will impact most people, most of the time, with something important to their lives as enterprise customers. Developer productivity tools are essential, but that topic is well covered elsewhere. The learnings also apply to that space; we won't use any examples or case studies from developer productivity tools. We will start by discussing the science and art of good design in the next section.

Understanding the science and art of UX design

Every coin has two sides (okay, plus the edge!). Typically, we see two sides to UX design. Those with visual backgrounds and those who come from a science perspective. Schools are now overwhelmingly delivering visual and graphic artists to meet demand, and with conversational AI, there is some *art*

to the experience but only sometimes visual elements. The introduction of generative AI impacts every facet of interaction design. The design roles will adapt or die. Adaption is the better option. As **graphical user interfaces** (**GUIs**) adapt to include conversational elements, the role of a visual designer will still be relevant, even if only to create the correct prompts to help them generate the look and feel that aligns with the organization's goals. The side of the equation for the science of design remains vital to requirements, understanding, and communication. Even when writing this book, ChatGPT provides some good answers related to UX. But what we cover in this book is not quickly answered by ChatGPT or any generative solutions. They help us, like all tools, move our design culture forward, but they don't know when they are wrong and still need us to decide where to apply the solutions, gather the correct data to help them form answers, and understand and improve the results.

The science of design

"*Anyone can design*," "*Just put the button there*," "*I can write this copy*." There is a difference between designing and making something. Anyone can make something. It might or might not work; it could work for some and not others, "*I designed this for myself, and I don't have any issues with it*," or it could be good. We want to use the tools, expertise, wisdom, and field knowledge to ensure *design decisions* yield the highest quality product. There is a wealth of research that usually underpins quality interactions, and we want to avoid pitfalls.

When we mean research, we include controlled studies with human subjects where the team has undergone rigorous processes to return reliable, repeatable, and valid results. We then take these results and apply them to our situation. And some will say, "*That doesn't apply to this because it is different*." Well, it could be, but that is why we share these results with interaction designers to guide us to what is applicable. As ChatGPT grows and integrates with other products and features, it will become more intertwined with visual elements, forms, interactive charts and visualizations, and even typical GUIs (with buttons, tables, filters, tabs, and all the components we see in any mobile, desktop, web, or embedded experience). This will make the science and historical expertise of UX design even more critical.

Let's take an example—Hick's Law; designers know and use Hick's Law all the time. "*The time it takes to make a decision increases as the number of alternatives increases*." This is why we have menus broken up into small segments, wizards for complex processes, and debate how many buttons should appear in a dialog box. In conversational flows, we keep decisions simple to reduce the burden on the user.

Hick-Hyman Law

This law was published in 1952 in the *Quarterly Journal of Experimental Psychology*. It is an equation, $RT = a + b\,log_2(n)$, where the response time (RT) is a function of the time not included in the decision-making (a) plus a constant (about 0.155 seconds) times the *log* function of the number of alternatives to choose from (n).

Article: `Wikipedia's explanation of Hick-Hyman Law (https://en.wikipedia.org/wiki/Hick%27s_law)`

We don't expect you to know or memorize this, but it is just one example of the science behind UX design decisions. Sometimes, knowing the guidelines, laws, and science helps you make better decisions and avoid mistakes by others, which you must learn to correct.

In this case, we know that a long list of choices is complex for users, and when a generative AI returns 10 to 15 choices, the effort it takes to decide goes up significantly. With this example, we can get these choices grouped into smaller logical segments and reduce them to two less complex decisions in a series. This is why we have the **File**, **Edit**, **View**, **Window**, and **Help** menus. By grouping menu items, picking the action is a less complex decision. It is also why menus fail when there are too many choices and no clear understanding of the organization of those items. Let's tell ChatGPT to return large decision trees as more logical and organized segments. We will cover this in *Chapter 7*, *Prompt Engineering,* to give ChatGPT instructions on how we want our responses framed.

How about one more classic example? Many phone calls to a business result in a voice prompt with 5-6-7-8 or even nine choices; how does a caller keep track of the right one? Do you ever have to listen to a prompt again? Maybe you got distracted and can't recall the first few choices. Do you ever use a few fingers to represent a number to remind yourself which answer might be best when multiple options are viable? This is a human working memory issue, a classic design problem.

Article: `Wikipedia explains working memory` (`https://en.wikipedia.org/wiki/Working_memory`)

These human factors impact the design of many experiences—especially ones based on a lot of text. We are not going to calculate Hick's Law in this book or test on working memory. Still, one should appreciate that applying design principles will be the cornerstone of helping create a successful ChatGPT experience. Without guidance, no one should be prompted with 35 choices on the first menu. This is an unacceptable user experience. So, we could use an existing and well-organized tree and have ChatGPT (speech-to-text) determine the customer's request to skip a few levels in one step.

Design book wish list

If you are new to design and want to learn the fundamentals, there is a wealth of wonderful resources. I suggest a few non-technical first books, such as Don Norman's *The Design of Everyday Things* or Steve Krug's *Don't Make Me Think*.

Those who are familiar with these works will want more sophisticated books. I suggest Jeff Johnson's book *Designing with the Mind in Mind* to help you understand how the fundamentals of psychology are used to derive many of these guidelines and thus help you apply these principles. *Universal Principles of Design* by Lidwell, Holden, and Butler is an excellent reference book. More resources are on our book list.

Website: `Recommended Book List` (`https://uxdforai.com/references#C13`)

To make these experiences successful, think like a designer. Consider how users will interact, use their expectations, biases, and assumptions, and how their unique experiences will shape their

future interactions. The power of the design mindset is to learn how to ensure people who use your product succeed.

To be clear, this is very different from *I will know it when I see it*. You want to apply art and style to the product and understand the scientific underpinnings of success. Some of this comes from psychology and related disciplines. Humans' cognitive and physical abilities have been unchanged over the last 50 years of the computer age. We still process information using the same senses. Our ability to react and respond is the same; only our experiences have evolved. We might now know how to type with two fingers, but there are limits to processing information, clicking on small targets, making decisions, or learning complex patterns. For example, just because today's cars have two to five times the horsepower of vehicles in the 1960s doesn't mean we can react any faster in an accident. As ChatGPT provides insights or understandings, we can overcome some cognitive or physical human limits using good UX practices.

The art of design

I bet you were waiting for us to talk about visual design here. This will apply in some places but won't be a book's cornerstone. We could have covered how to design a chat window or what visual treatment works best, but I suspect that information is in many resources independent of a book on ChatGPT. Brand identity has its subjective standards and its collection of science and expertise. Now, there will be experiences that join generative AI and traditional user interfaces, and one can apply a visual style and expertise to that experience, but that is unique to the enterprise. We want to cover the art of design that can span any enterprise here and join this with the science that supports design. And all of this is done to create something expected and comfortable for the audience.

In meetings, we hear, "*Put on your designer hat*." This means being a customer's advocate and addressing customer's needs through how they interact with the product. This is within the basics of user-centered design. We bring customers into the process because we sometimes need to learn better. Listening to the customer means something different than what the customer says is right. It isn't, and it won't be. Many times, what the customers ask for and what they need are different. They want to work around a problem, but maybe the solution is to eliminate the problem in the first place. This is why this is also an art. You have to know where to look for problems worth solving.

Engaging users in the process will help us understand expectations and see how designs impact their behavior. And we know that generative AI will only sometimes give the expected results. Part of the designer's job is to improve this interaction, likely indirectly, through the tools we will discuss in the book, such as data cleaning, prompt engineering, fine-tuning, user research, user testing, and monitoring. There is also art we need to control and refine in generating responses with AI. As Kevin Mullet explained in a conversation for this book, "*The science describes how things ought to be done, while the generative AI describes (ideally) how they are typically done today. The designer's task (as always) is to wrangle those diverse inputs into a single coherent solution that best maps to the user's wants and needs*." So, we need to use our depth of skills to keep our ChatGPT solutions on point.

To be fair to the process, only some decisions can be based on logic or a research method, especially regarding how you want ChatGPT to respond. Let's break down the art concept as it applies to the words used in a product's voice, style, and tone.

Let's take a large company selling surfing apparel, surfboards, and beach equipment. The website or style should be distinct from a stodgy, old-fashioned bank. Right, dude, and *dudettes*? Likewise, a bank intends to come across as reliable, safe, dependable, and trustworthy, Mrs. Customer. So, the site should speak to customers in a manner consistent with their expectations for the brand.

The overall *voice* of your conversations should reflect the brand.

The *style* of interactions should match the users' needs, wants, and expectations.

And the *tone* of the conversation can adapt or vary based on the current situation.

We will address these in detail to identify when and how to vary the tone within a style for the overall voice of a ChatGPT assistant.

There is no one correct answer to how to write. The voice, style, and tone can certainly overlap in concept. The same thing can be said in many ways and still be understood:

```
Your appointment is now scheduled for this coming Monday.
I have scheduled you to see the doctor on Monday.
Confirmed. I have a Monday appointment in the books.
```

We will discuss how to ensure that responses are clear, concise, and contain the right level of detail (maybe the customer wondered when the appointment was? Or maybe at what office?). Or perhaps they expected a calendar (`.ical` or similar) attachment to make it easier to add this appointment to a calendar:

```
Got it. Monday at 3:00 pm at the Palo Alto office. Here is the
calendar entry.
```

If a manager scheduled this appointment at the surf shop we mentioned, they would use lingo and tone that fits the customers to make the customer feel like one of them:

```
Akaw! We got you, grom. Swing by for your intro lesson on Mondo at 3
at the Santa Monica beach shop.
```

An intelligent designer will know that even if the audience doesn't know *Akaw* (awesome or cool) and *grom* (a new surfer), they can still understand the message and be a little excited to get their surf on.

Use cases for this book

This is where the art and science of design meet. And we will see this a lot in our journey. We will demand that our ChatGPT instance be clear, complete, and conversational (when needed). Using ChatGPT to create answers that present themselves in a non-conversational user interface is also

perfectly reasonable. ChatGPT can process and generate data in a backend system. A UI can use alerts, buttons, or a warning dialog supplemented by ChatGPT. We call that a hybrid user experience. A combination of traditional UI elements and generative AI. Grammar or style suggestions when writing an email could be made in a suggestion window, and it doesn't have to be conversational. We will refer to these as recommenders. They provide recommendations to assist in content generation or to influence a process, such as a sales lead.

Let's illustrate these concepts with pictures. Each gives a sense of the experiences covered in the book's examples. If your use case differs but will still be a user interface, then most of what we cover will likely apply.

Messaging

Simple text messaging (**SMS**), voice experiences, or simple chats are two-way user experiences that can only use text or speech. These experiences have limited interactions in SMS: text, images, file uploading, and links. *Figure 1.3* shows an example of interacting with my expense assistant on the phone via SMS. It is easy to access and simple but has limits to the types of interactions it can support.

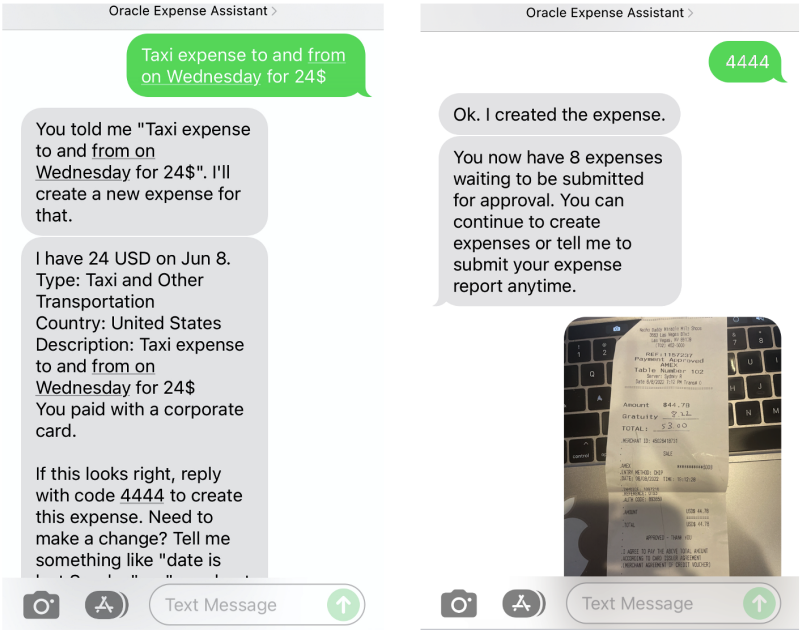


Figure 1.3 – An example of simple messaging via SMS on a mobile phone

If this was a voice experience, like in a car, or a skill in Amazon Alexa, creating well-done voice-only solutions is even more challenging than with SMS. We will discuss voice experiences in the book, but we should discuss Hybrid UIs next, as that is the future.

Hybrid UIs

Slack, Teams, and web interfaces can incorporate user interface elements with conversational text. Simple experiences can have links, buttons, charts, graphs, or forms. This hybrid experience combines LLMs and GUIs, allowing for complexities not quickly addressed by generative text or GUI components alone. In *Figure 1.4*, I show a simple example from Slack where buttons encourage the exploration of tasks.

Article: `ChatGPT AI for Slack` (`https://www.cityam.com/slack-to-offer-users-a-chatgpt-ai-tool-which-will-write-messages-for-them-in-seconds/`)

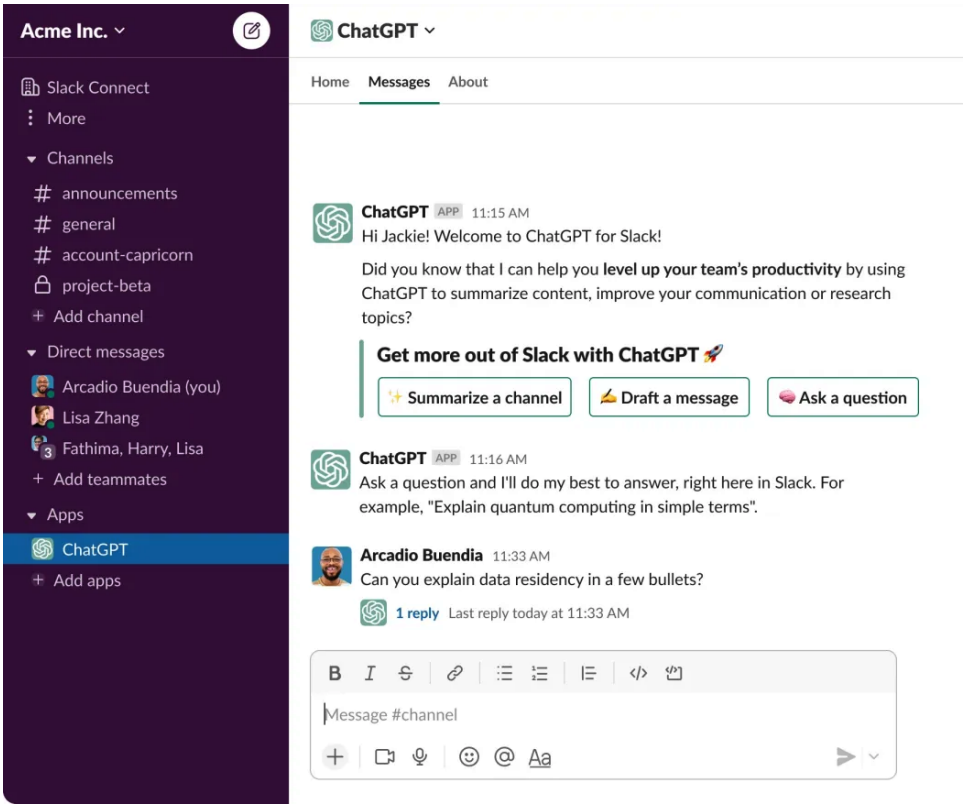


Figure 1.4 – An example of a hybrid user experience

Overall, experiences in more robust channels can take advantage of even more creative elements. For example, a web channel might support interactive charts or visualizations, shopping cart items, information, tools, and products from the enterprise. Generative conversations can refine the results or change perspective or filters, while UI components also control aspects of the view. This is by far the most robust and creative space for generative solutions.

Recommender UIs

Typically, a recommender is a recommendation as a textual prompt to encourage a specific behavior. It could be done with action buttons, but the user does not interact or converse with a system. A summary or writing suggestion tool is typically this kind of experience. Even if it has buttons to "generate email," if the user is not conversationally interacting with the generative agent, it is more like a recommender. In this Salesforce example in *Figure 1.5*, Einstein gives context to the sales process.

Article: `AI in Salesforce (https://www.salesforce.com/products/ai-for-sales/)`

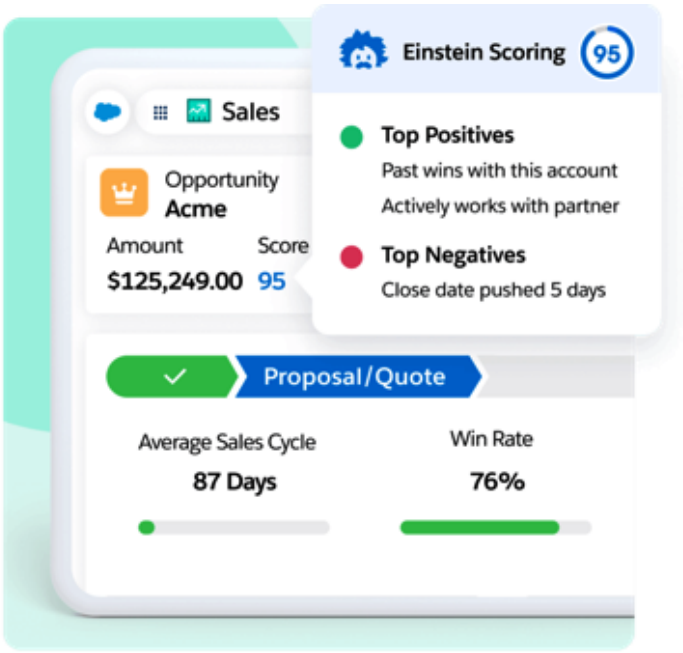
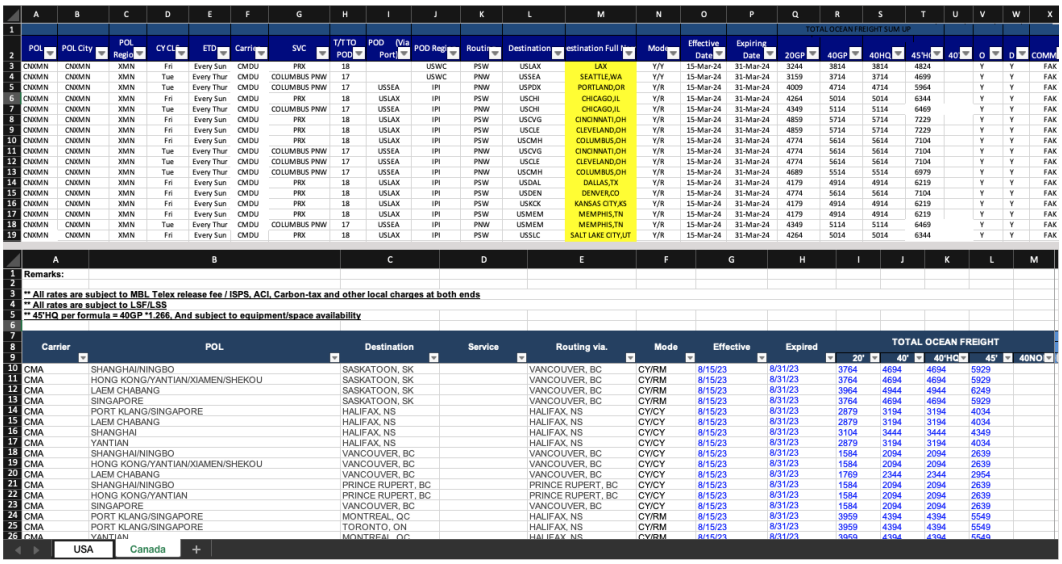


Figure 1.5 – An example of a recommendation UI

The backend uses for ChatGPT

A hidden use of a generative AI solution can be to process information and provide data, information, or wisdom to the user that is then passed to a different user experience. The user doesn't directly interact with the generative AI, so a UI doesn't exist. There could be administrative UIs or UIs that use the processed results. In *Figure 1.6*, we don't see a user interface. We see data that would be ingested, processed, and then analyzed to return clean, normalized data. We will explore this example in a

case study starting in *Chapter 6*, *Gathering Data – Content is King*. Just realize we don't have to have a UI to get much value out of ChatGPT and generative solutions; we might use it as a tool in a much larger process.



POL	POL City	POL Regio	CY CLS	ETD	Carrie	SVC	T/T TO POD	POD	(Via Port	POD Regi	Routi	Destination	estination Full N	Mod	Effective Date	Expiring Date	20GP	40GP	40HQ	45'H	40'	O	D	COMM
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18			USWC	PSW	USLAX	LAX	Y/Y	15-Mar-24	31-Mar-24	3244	3814	3814	4824		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17		USSEA	IPI	PNW	USSEA	SEATTLE,WA	Y/Y	15-Mar-24	31-Mar-24	3159	3714	3714	4699		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17	USSEA		IPI	PNW	USPDX	PORTLAND,OR	Y/R	15-Mar-24	31-Mar-24	4009	4714	4714	5964		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USCHI	CHICAGO,IL	Y/R	15-Mar-24	31-Mar-24	4264	5014	5014	6344		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17	USSEA		IPI	PNW	USCHI	CHICAGO,IL	Y/R	15-Mar-24	31-Mar-24	4349	5114	5114	6469		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USCVG	CINCINNATI,OH	Y/R	15-Mar-24	31-Mar-24	4859	5714	5714	7229		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USCLE	CLEVELAND,OH	Y/R	15-Mar-24	31-Mar-24	4859	5714	5714	7229		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USCMH	COLUMBUS,OH	Y/R	15-Mar-24	31-Mar-24	4774	5614	5614	7104		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17	USSEA		IPI	PNW	USCVG	CINCINNATI,OH	Y/R	15-Mar-24	31-Mar-24	4774	5614	5614	7104		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17	USSEA		IPI	PNW	USCLE	CLEVELAND,OH	Y/R	15-Mar-24	31-Mar-24	4774	5614	5614	7104		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17	USSEA		IPI	PNW	USCMH	COLUMBUS,OH	Y/R	15-Mar-24	31-Mar-24	4689	5514	5514	6979		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USDAL	DALLAS,TX	Y/R	15-Mar-24	31-Mar-24	4179	4914	4914	6219		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USDEN	DENVER,CO	Y/R	15-Mar-24	31-Mar-24	4774	5614	5614	7104		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USKCK	KANSAS CITY,KS	Y/R	15-Mar-24	31-Mar-24	4179	4914	4914	6219		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USMEM	MEMPHIS,TN	Y/R	15-Mar-24	31-Mar-24	4179	4914	4914	6219		Y	Y	FAK
CNXMN	CNXMN	XMN	Tue	Every Thur	CMDU	COLUMBUS PNW	17	USSEA		IPI	PNW	USMEM	MEMPHIS,TN	Y/R	15-Mar-24	31-Mar-24	4349	5114	5114	6469		Y	Y	FAK
CNXMN	CNXMN	XMN	Fri	Every Sun	CMDU	PRX	18	USLAX		IPI	PSW	USSLC	SALT LAKE CITY,UT	Y/R	15-Mar-24	31-Mar-24	4264	5014	5014	6344		Y	Y	FAK

Remarks:

**** All rates are subject to MBL Telex release fee / ISPS, ACI, Carbon-tax and other local charges at both ends**
**** All rates are subject to LSF/LSS**
**** 45'HQ per formula = 40GP *1.266, And subject to equipment/space availability**

Carrier	POL	Destination	Service	Routing via.	Mode	Effective	Expired	TOTAL OCEAN FREIGHT 20'	40'	40'HQ	45'	40NO
CMA	SHANGHAI/NINGBO	SASKATOON, SK		VANCOUVER, BC	CY/RM	8/15/23	8/31/23	3764	4694	4694	5929	
CMA	HONG KONG/YANTIAN/XIAMEN/SHEKOU	SASKATOON, SK		VANCOUVER, BC	CY/RM	8/15/23	8/31/23	3764	4694	4694	5929	
CMA	LAEM CHABANG	SASKATOON, SK		VANCOUVER, BC	CY/RM	8/15/23	8/31/23	3964	4944	4944	6249	
CMA	SINGAPORE	SASKATOON, SK		VANCOUVER, BC	CY/RM	8/15/23	8/31/23	3764	4694	4694	5929	
CMA	PORT KLANG/SINGAPORE	HALIFAX, NS		HALIFAX, NS	CY/CY	8/15/23	8/31/23	2879	3194	3194	4034	
CMA	LAEM CHABANG	HALIFAX, NS		HALIFAX, NS	CY/CY	8/15/23	8/31/23	2879	3194	3194	4034	
CMA	SHANGHAI	HALIFAX, NS		HALIFAX, NS	CY/CY	8/15/23	8/31/23	3104	3444	3444	4349	
CMA	YANTIAN	HALIFAX, NS		HALIFAX, NS	CY/CY	8/15/23	8/31/23	2879	3194	3194	4034	
CMA	SHANGHAI/NINGBO	VANCOUVER, BC		VANCOUVER, BC	CY/CY	8/15/23	8/31/23	1584	2094	2094	2639	
CMA	HONG KONG/YANTIAN/XIAMEN/SHEKOU	VANCOUVER, BC		VANCOUVER, BC	CY/CY	8/15/23	8/31/23	1584	2094	2094	2639	
CMA	LAEM CHABANG	VANCOUVER, BC		VANCOUVER, BC	CY/CY	8/15/23	8/31/23	1769	2344	2344	2954	
CMA	SHANGHAI/NINGBO	PRINCE RUPERT, BC		PRINCE RUPERT, BC	CY/CY	8/15/23	8/31/23	1584	2094	2094	2639	
CMA	HONG KONG/YANTIAN	PRINCE RUPERT, BC		PRINCE RUPERT, BC	CY/CY	8/15/23	8/31/23	1584	2094	2094	2639	
CMA	SINGAPORE	VANCOUVER, BC		VANCOUVER, BC	CY/CY	8/15/23	8/31/23	1584	2094	2094	2639	
CMA	PORT KLANG/SINGAPORE	MONTREAL, QC		HALIFAX, NS	CY/RM	8/15/23	8/31/23	3959	4394	4394	5549	
CMA	PORT KLANG/SINGAPORE	TORONTO, ON		HALIFAX, NS	CY/RM	8/15/23	8/31/23	3959	4394	4394	5549	
CMA	YANTIAN	MONTREAL, QC		HALIFAX, NS	CY/RM	8/15/23	8/31/23	3959	4394	4394	5549	

USA | Canada | +

Figure 1.6 – Spreadsheets processed in a backend ChatGPT solution

I suspect these four concepts cover almost all uses of textual generative AI. The most interesting for us are ones with richer or robust UX, while something like a pure backend solution leaves little discussion to improve the user experience. But we do have examples! Sometimes, work starts as a backend solution, and then, with the need for controls and feedback mechanisms, these experiences come to the forefront. None of this work can be done in isolation. It takes a team.

It takes a village to create superb UX

If you are serious about creating world-class experiences (and I hope every reader is!), consider the resources that will help get the most value on the design side. As shown in *Figure 1.7*, the collection of specialists you need goes beyond the typical software team. Let's explore their contributions briefly, as this is not the usual team organization for traditional software development.

Figure 1.7 – It takes a village to build a solution

Writing comes into play fundamentally because we have an assumption in the enterprise design space that there are materials, FAQs, articles, manuals, help, installation processes, and bug fixes, which might not be easily accessible to the Internet. This material will be added to a private ChatGPT instance so that a paying customer will get value. We hope the business employs writers and editors to create high-quality content. How this is written, ingested, and represented in ChatGPT will reflect strongly on how well it was written originally (quality in will help with quality out). Designers must help understand the goals, design tools, and even integrations that might optimize the user's experience. Should a customer answer 15 questions individually or use a form to see the chunks of questions and answers contextually? We must design the experiences to match the problem we need to solve (the use case). In complex environments, where technical language or even multiple written or verbal languages are expected, linguists will be part of the team to help train and improve communication. Writers and linguists are also crucial in setting the style and tone of the conversation. We will dive into this in more detail later.

Finally, let's not leave off *user research*. Besides reviewing logs and analyzing the product's use, researchers can work with customers to learn more about how they want to use the assistant, where and when they will likely call on it, and how they will interact with it. These learnings are valuable for plotting a strategic direction and can help fix tactical issues. We will provide a set of heuristics that can be used to put on your research hat to understand where problems will arise and how to classify them.

This is just on the design side. We know that a team of engineers, product managers, quality engineers, and others will be involved in the journey. They can also benefit from this book. Do share it. Great ideas can come from anywhere, so always keep learning. The best way to keep learning is not to relearn something we already know from the history of conversational AI.

So, we now know who should be involved in this process, and we understand that OpenAI's ChatGPT is foundational, but there is still one missing piece. Why is there a need for a custom version of ChatGPT? Because proprietary data and custom answers are unavailable in a worldly-trained ChatGPT instance. Let's explore how to add unique content by setting up a personalized ChatGPT instance.

Setting up a customized model

So now we have a sense of the people and tools we need. One more piece of the puzzle is the content that will make a ChatGPT solution valuable. There are many ways to include content in an LLM, such as ChatGPT. However, we assume a focus on unique, proprietary content hidden behind a secure paywall or available only to authenticated users. If company answers are already out in the world and answered by the basic ChatGPT, ask yourself, *How would my custom version of ChatGPT provide value?* We can help get that answer by focusing our content discussions on building a private model and including company data without sharing it with the world.

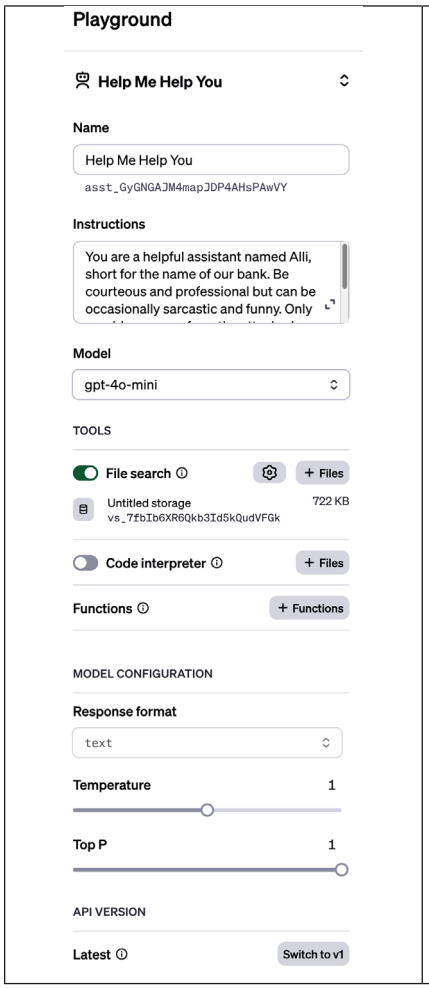
The idea of *enterprise* assistants is precisely that. Ensure company data is only exposed to customers and that adding it to an instance of ChatGPT does not expose the data to the world. Only some people were this careful when ChatGPT was first introduced. Don't add sensitive data to a model used by competitors or to be a doctor and accidentally break patient confidentiality. The enterprise market demands security. Look at this article to go a little deeper:

Article: `Feeding Sensitive Data to ChatGPT (https://web.archive.org/web/20240119052608/https:/www.darkreading.com/cyber-risk/employees-feeding-sensitive-business-data-chatgpt-raising-security-fears)`

Let's make a simple request and see how adding data dramatically changes the landscape for answers. I will set up a ChatGPT assistant in the Playground for this example. I am selecting the gpt-3.5-turbo-1106 model because it supports "Retrieval" and allows me to upload my files. Retrieval means I can share content with the AI so it can answer based on the information I provide. This sets enterprise solutions apart from our day-to-day use of public conversational assistants. An enterprise solution knows things that are specific to the enterprise. For this example, everything can be done from the Playground user interface. We don't need to set up or run any code. You will need a free account to run this. Don't upload content you are not comfortable sharing.

Newer models, like Chat GPT 40-mini, are always becoming available; use the latest models in your playground. In my example, I included a 200-page document (a draft of my Master's thesis on interface design). This material is not easily accessed online, so we can see how results change based on its inclusion.

Uploading this document took only a few seconds, and the model was ready to answer. Of course, adding thousands or a million files is very different. Still, I want to share a sense of this power and ChatGPT's innate ability to prioritize uploaded content over the base model. Follow along using my example content, as shown in *Figure 1.8*.



1. Sign up for an account.
2. Website: `OpenAI https://chat.openai.com/`
3. Head over to the Playground.
4. Website: `OpenAI Playground https://platform.openai.com/playground`
5. Choose **Assistants** from the top menu.
6. Select a model that supports Retrieval.
7. Test the model by asking question before uploading documents.
8. Create **Instruction** (this is prompt engineering) to ask it to "focus on answers only gathered from the documents".
9. Upload documents and retest – asking questions that only are known within the document.

Figure 1.8 – Setting up the Playground to do a simple test

We can compare the answer provided by the base model to the same question answered after adding the `Full Thesis.pdf` file, a draft of almost 200 pages.

GitHub: `Chapter1-Full_Thesis.pdf (https://github.com/PacktPublishing/UX-for-Enterprise-ChatGPT-Solutions/blob/main/Chapter1-Full_Thesis.pdf)`