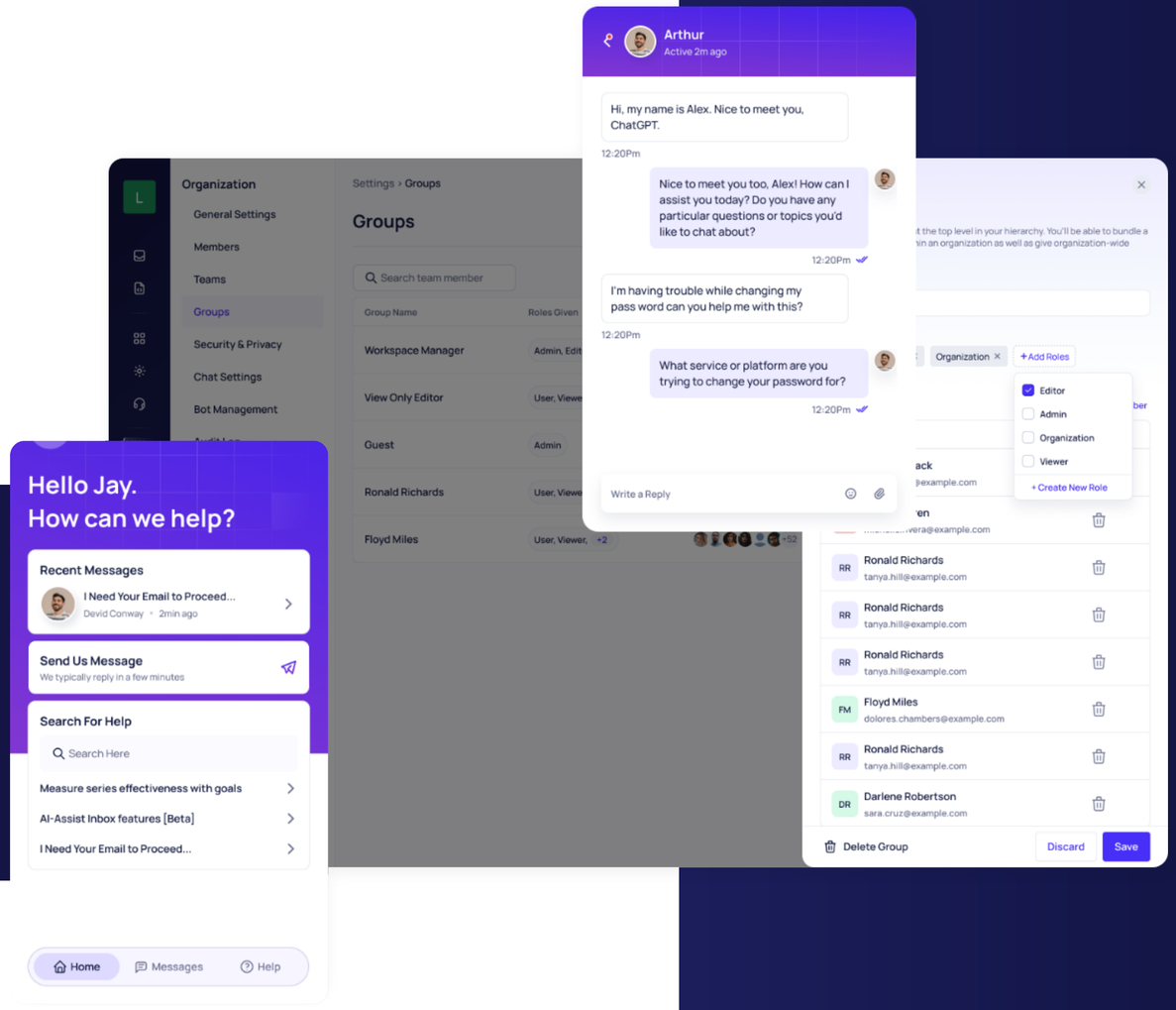
BizGroup
INNOVATING TOGETHER

An AI-Powered Chatbot

To Improve Customer Engagement

We helped our client with development and assisted in clocking a $24 million market valuation.

NEXT →



Get to Know

The Product

The AI-driven customer service chatbot is your ultimate solution for authentic, human-like communication. With seamless integration and dynamic responses, it's like having a knowledgeable team member available 24/7. The chatbot learns from human agents, ensuring personalized interactions and reducing agent inboxes. From refunds to subscription changes, it adapts effortlessly to meet customer needs. Experience the efficiency of automated conversations without sacrificing the human touch. The chatbot aims to reduce workloads and empower businesses to align their team with other market-competitive tasks.

Problems

AI Chatbot Solves

The client encountered significant challenges in the customer service industry. Later, the client decides to bridge the gap with the help of AI. In earlier days, keeping your support active 24/7 requires an increased workforce. The other side of the coin was that various gaps were still untouched and extended beyond mere responsiveness. Overcoming the challenge of delivering an enhanced solution and experience to the customers, the client aimed.

Real-Time AI Training:

Update information in real-time for continuous learning.

Refunds:

Streamline the process by requesting reimbursement.

Payment Resolution:

Resolve payment issues or raise a dispute quickly.

Subscription:

Manage, change, or upgrade your subscription plan per your needs.

Policy Clarification:

Know the policies for claiming a coupon to raise an issue.

Our

Development Process

Our task involved collaborating with product managers, developers, and business analysts to craft an optimal chatbot for customer support. Working closely together, we synthesized insights and expertise to develop a robust solution tailored to meet the diverse needs of our client and their customers. For the beta version, we selected 8 participants to test and stay unbiased on the product outcomes.

Conceptualize:

We did Market research, user feedback, and competitive analysis for developing innovation.

Design:

We created seamless experiences through user flow, journey mapping, and wireframing.

Development:

We worked on building for Android and iOS through rigorous user testing.

Deployment:

We launched the AI Chatbot on Android and Apple stores.

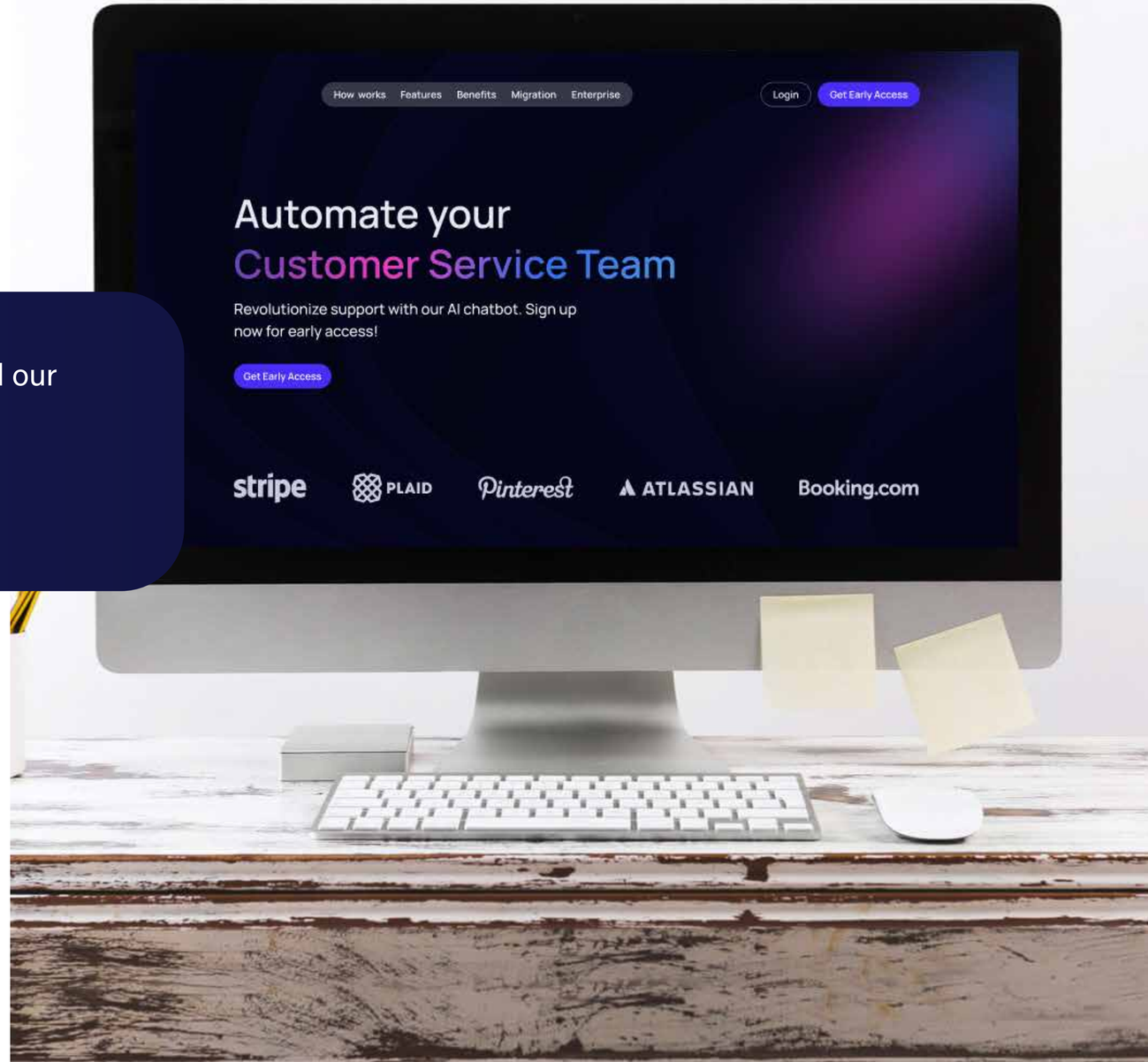
The results of our product in the beta testing phase helped us address the even odds. 8 out of 8 participants found the idea useful to offload repetitive tasks with the help of Generative AI. During the beta testing, some points still need to be in consideration, and empowering businesses to leverage AI technologies. After months of working on the gaps, the final output brought a spark of happiness to the client's face. Let's take a moment to read his words.

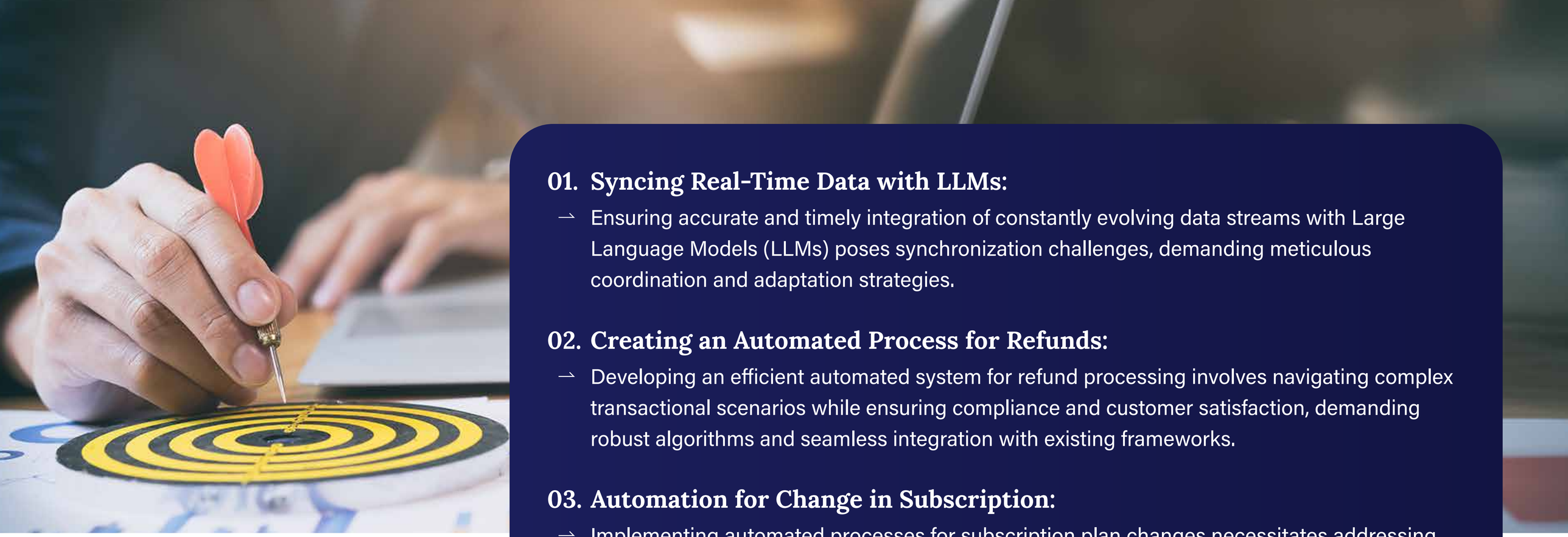
From The

CEO Desk

"Their software development skills are inspiring. They quickly converted our complex challenges into a solution. Also, their UI/UX team has done an awesome job."

~ Andrew





Project Challenges

The project challenges encountered during software development demanded innovative solutions and strategic problem-solving for successful outcomes.

01. Syncing Real-Time Data with LLMs:

- Ensuring accurate and timely integration of constantly evolving data streams with Large Language Models (LLMs) poses synchronization challenges, demanding meticulous coordination and adaptation strategies.

02. Creating an Automated Process for Refunds:

- Developing an efficient automated system for refund processing involves navigating complex transactional scenarios while ensuring compliance and customer satisfaction, demanding robust algorithms and seamless integration with existing frameworks.

03. Automation for Change in Subscription:

- Implementing automated processes for subscription plan changes necessitates addressing diverse user preferences and billing structures, requiring sophisticated algorithms and seamless integration with backend systems to ensure accuracy and user satisfaction.

04. Getting LLMs to Understand Policy's Rational Aspect:

- Teaching Large Language Models (LLMs) to comprehend and apply the rational aspects of policies involves overcoming semantic nuances & contextual ambiguities, demanding advanced Natural Language Processing (NLP) techniques & iterative refinement strategies.

Biz4Group's

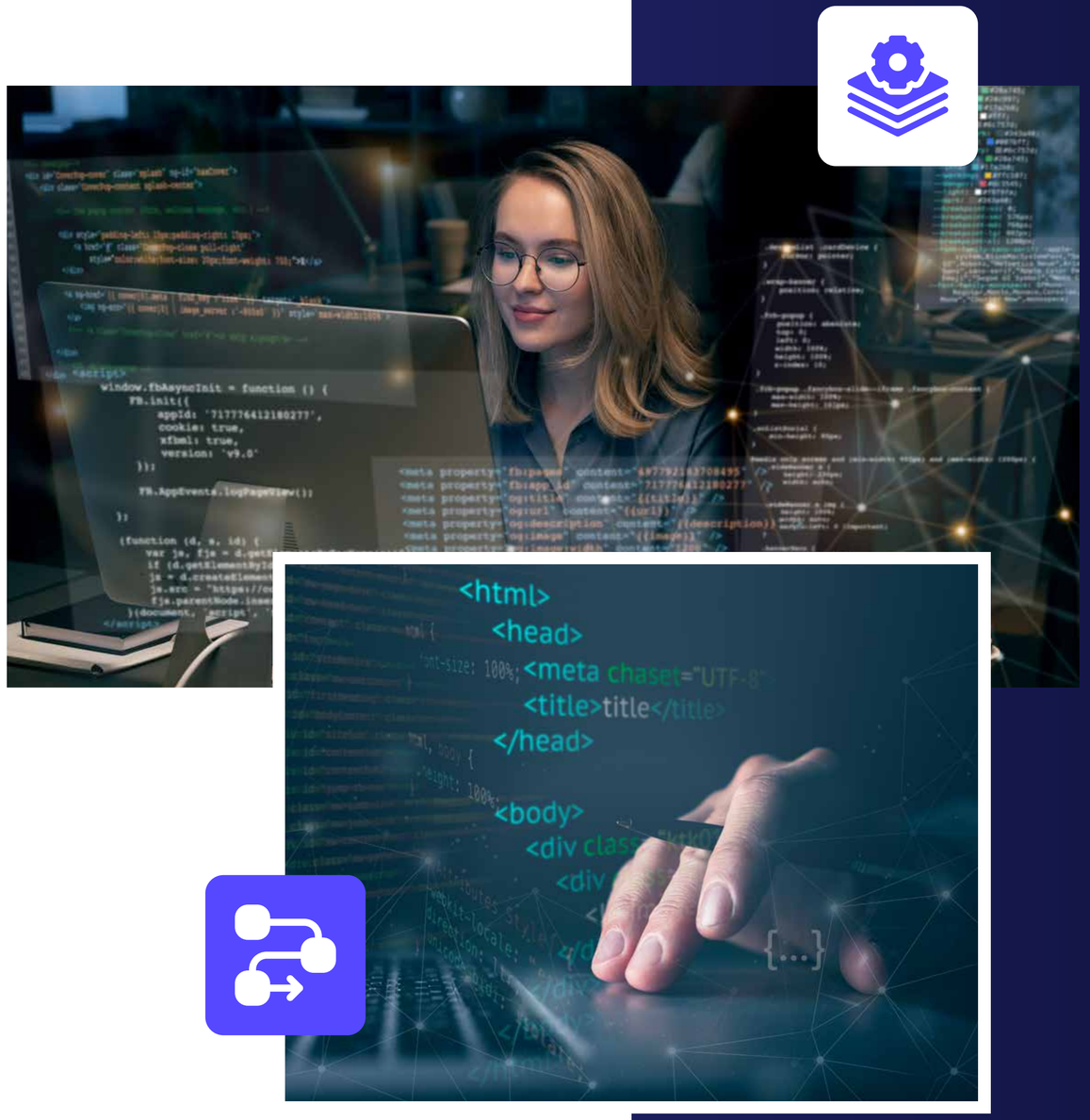
Development Process

We embarked on our mission to pick the best from our talent pool to prioritize accessibility, instant connectivity, and efficiency. Our journey commenced with a seven-day interactive session where the team understood the client's vision and desired features.

Drawing insights from this collaborative sprint, we crafted a comprehensive solution to streamline the entire process. This involved everything from identifying key market metrics to refining structures and prospects.

Moreover, our team spearheaded several integration features, empowering businesses to connect with customers proactively when necessary. This fusion of technology and human connection facilitates swift and meaningful interactions, bridging the gap between customers and agents.

Through our collaborative efforts and innovative solutions, we're reshaping the customer service landscape, enabling executives to invest their time in learning or boosting productivity on meaningful tasks.



Tech
Stack

Admin Panel:

React

User Panel:

Next JS

React

Backend:

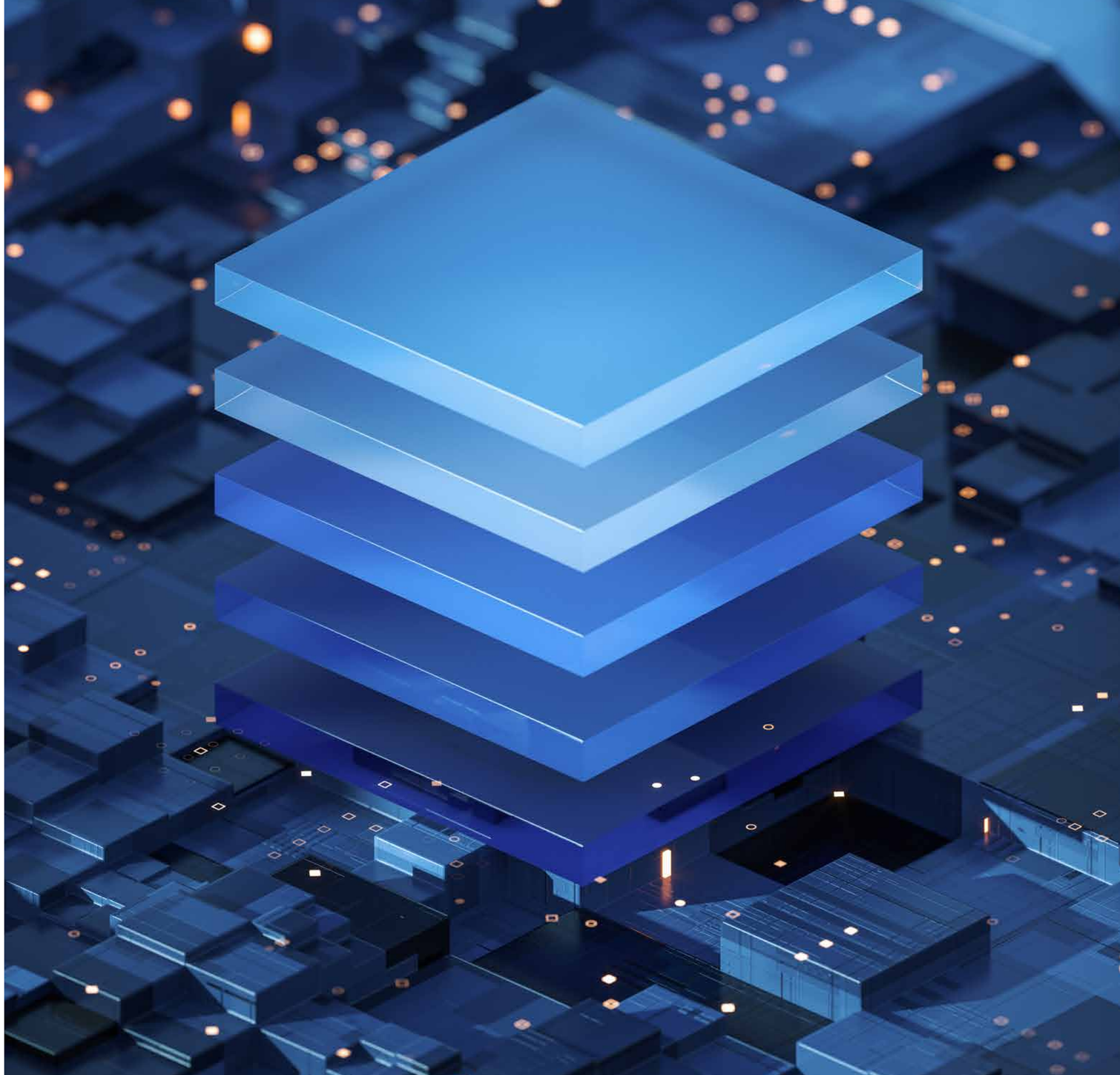
Python

FastAPI

Database:

PostgreSQL

Milvus

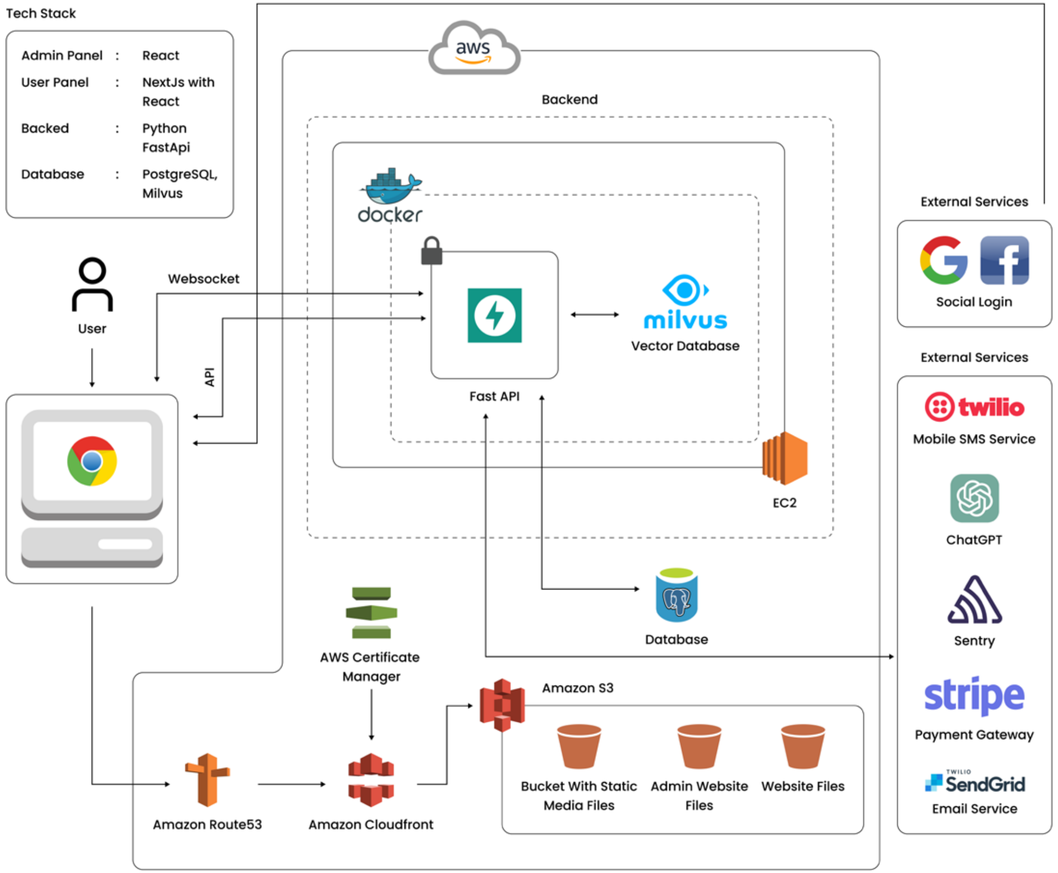


Technical

Architecture

This architecture diagram represents a web application infrastructure built on Amazon Web Services (AWS) and incorporates a mix of internal services and third-party integrations.

The user interfaces include an Admin Panel and a User Panel, both developed with React, a popular JavaScript library for building user interfaces. The Admin Panel is enhanced with Redux for state management, ensuring a predictable state container across the app. The User Panel is noted for being constructed with Next.js, a React framework that enables server-side rendering and generates static websites.



The backend logic of the application is powered by Python with FastAPI, a modern, fast (high-performance) web framework for building APIs with Python based on standard Python-type hints. The backend services run inside Docker containers, which provide a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, libraries, and settings.

For data storage, PostgreSQL is employed as the primary database, known for its robustness and reliability. Milvus, an open-source vector database built for AI applications, is used alongside PostgreSQL to handle complex data structures that are typical in AI-driven applications.

The WebSocket protocol is used for real-time bidirectional communication between the client and server, enhancing the application's interactivity.

On the AWS side, the services used include:

- **EC2 (Elastic Compute Cloud)** for hosting the application servers.
- **S3 (Simple Storage Service)** for storing static media files, admin website files, and website files.
- **Route 53** for domain name system (DNS) management.
- **CloudFront** as a content delivery network (CDN) to distribute the content quickly and securely.
- **Certificate Manager** for managing SSL/TLS certificates.

External services integrated with the system:

- **Google and Facebook** for social login capabilities.
- **Twilio** for mobile SMS services.
- **ChatGPT** for conversational AI.
- **Sentry** for real-time error tracking.
- **Stripe** for payment gateway services.
- **SendGrid** for email services.

This infrastructure provides a secure, scalable, and performant foundation for a modern, AI-driven web application.

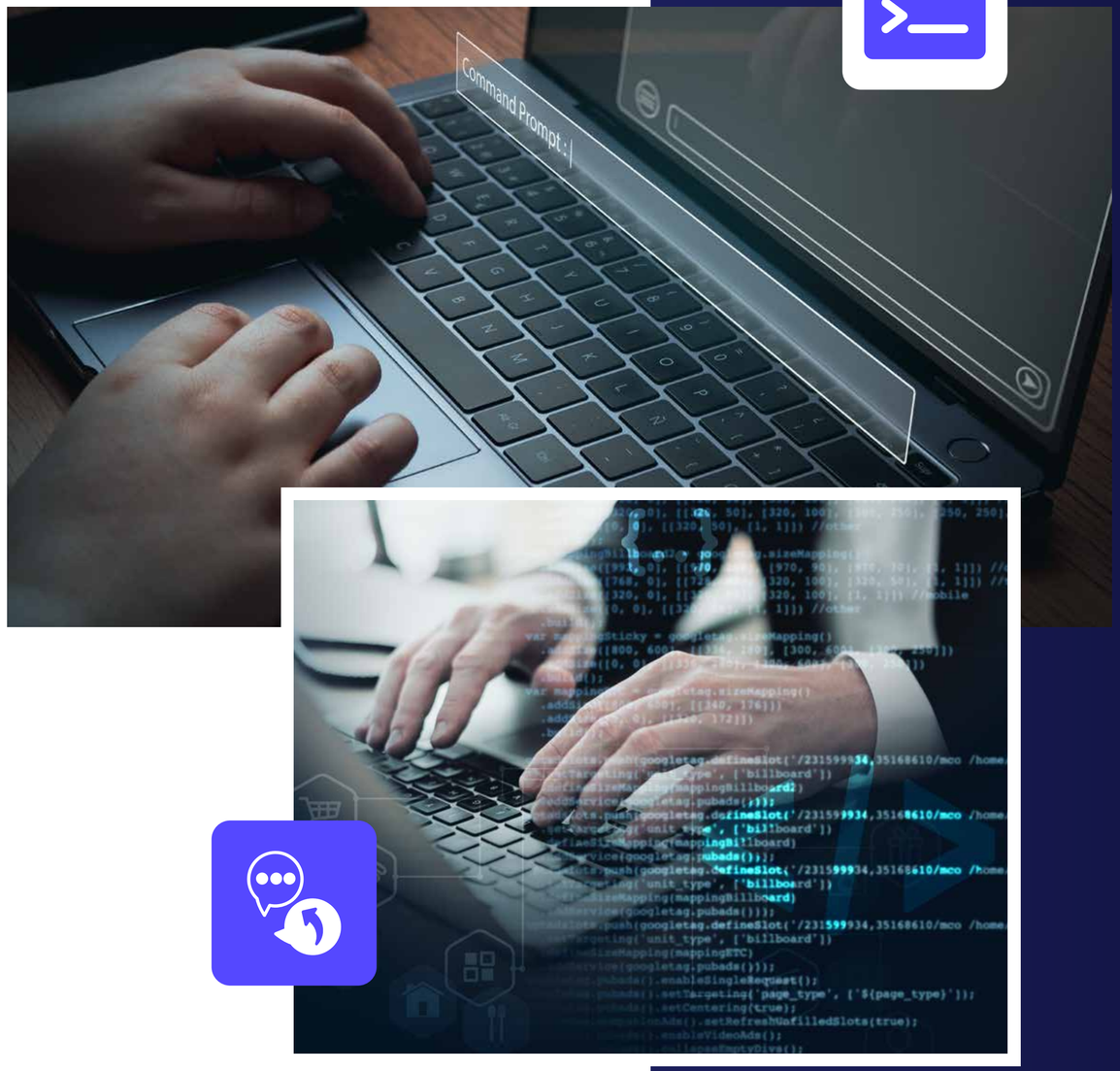
Request Processing Module (RPM)

When a user interacts with the system by entering a prompt, here's the journey the data takes within this architecture:

The user enters the prompt, either through the User Panel or the Admin Panel, through a web browser or a mobile application interface.

React, possibly with the help of Next.js, processes the user input. If it's an administrative action, Redux might update the application state accordingly. This processed input is then sent to the backend through a secure WebSocket connection or HTTP/S request.

The request from the user is routed through AWS Route53, which resolves the domain name to the correct IP address. If it's a request for static content, CloudFront, AWS's CDN, delivers it from the nearest edge location for lower latency.



The request reaches the backend services hosted on AWS EC2 instances. FastAPI, running within Docker containers, receives the request. It’s responsible for handling the business logic of the application, which might include interacting with the databases or external services.

Depending on the nature of the prompt:

- If the request requires transactional data operations, PostgreSQL is queried or updated.
- If the prompt involves AI functionalities, such as search or recommendation, Milvus, the vector database designed for AI applications, is used.

The backend may communicate with various external services to fulfill the request, such as:

- Authentication through Google or Facebook, if the prompt involves signing in or up.
- Sending SMS via Twilio if the action requires a mobile notification.
- Interfacing with Stripe for payment processing if the transaction is financial.
- Utilizing SendGrid for sending out emails related to prompt action.
- Sentry could capture any errors that occur during the process for real-time alerting and debugging.
- ChatGPT or similar AI services could be queried for natural language processing or other AI-driven tasks.

After the backend logic is executed and the necessary data is fetched or processed, FastAPI sends a response back to the user interface.

The response is rendered on the User or Admin Panel, providing the user with feedback, such as confirmation of an action, the outcome of a request, or the requested data.

The robust infrastructure ensures security, scalability, and performance while combining internal and external services enables a seamless, feature-rich user experience. The journey continues if any new session begins from the user's end.

Results

Sales:
$05 Million

Market Valuation:
$24 Million

Subsequently, the client underwent a share dilution process, resulting in the shifting of operational control to the appointed takeover director.

The

Future Overview

The outlook for AI chatbots in customer service is promising. With 23% of companies already utilizing them and 67% of consumers engaged with chatbots in the past year, adoption rates are expected to rise. Consumer preference for chatbots (74%) and the growing expectation for their presence on websites (73%) underscore their significance. Additionally, chatbots offer substantial cost savings (30%) and efficiency gains, with the ability to answer 80% of routine questions and handle full conversations 69% of the time. This technology shows potential in resolving various customer issues, further enhancing its value proposition.