ChatGPT

CareCircle: AI-Powered Family Care Orchestration Platform

Introduction

CareCircle is an AI-driven family care coordination platform designed for distributed families to collaboratively care for their elderly loved ones. It intelligently monitors for signs of cognitive or behavioral changes in everyday communications and streamlines the delegation of caregiving tasks among family members. By leveraging AWS cloud services (within Free Tier limits) and generative AI, CareCircle provides a **real-time, multilingual, and multi-agent** solution that helps families stay connected and proactive in elder care. The platform is built with AWS best practices and targets the AWS 10,000 AIdeas Hackathon's **Workplace Efficiency** and **Social Impact** tracks, delivering measurable value in reducing caregiver stress, preventing costly health incidents, and improving coordination.

Problem & Motivation

Modern families often live far apart, juggling full-time jobs while caring for aging parents. Over **53 million Americans (1 in 5 adults)** are family caregivers for an older relative [1], and the economic value of unpaid elder care is estimated between **$96–$182 billion annually** [2] . This vital care comes at a personal cost: on average caregivers spend **26% of their income** on caregiving expenses [3] , and many sacrifice work opportunities (12% reduce hours or leave jobs due to caregiving [4]). Early signs of dementia or health decline are often missed until a crisis occurs, leading to expensive hospitalizations or accidents (falls alone cost the U.S. ~$50–80 billion in healthcare annually [5]).

Existing solutions only partially address these challenges. Traditional caregiver apps (e.g. Lotsa Helping Hands) rely on manual task scheduling and lack intelligent insights [6] . Voice assistants like Amazon's Alexa Together offered basic remote monitoring but were discontinued in 2024 [7] , leaving a gap for tech-enabled family caregiving. Professional systems like Addison Care+ use virtual avatars and AI for 24/7 monitoring [8] [9] , but these are enterprise-focused, require special hardware, and can be costly. Families need a **comprehensive yet accessible** tool that proactively detects issues (especially cognitive changes), coordinates help across multiple family members, and respects cultural/language needs. CareCircle was conceived to fill this void – **preventing crises and caregiver burnout through AI orchestration**, while operating within the free, serverless AWS stack for affordability and scalability.

Solution Overview

CareCircle combines multimodal AI sensing with a coordinated task management hub to support aging in place. Key capabilities include:

- **Real-Time Behavioral Drift Detection:** The platform analyzes voice and video calls with the elder in real time to detect signs of memory lapses, confusion, or emotional distress. By using AI to monitor

conversation *tone*, *speech patterns*, and *keywords*, CareCircle can flag potential cognitive decline or urgent needs (e.g. senior sounding disoriented about medication). This early warning system gives families valuable lead time to intervene.

- **Multi-Agent Task Assignment:** CareCircle employs a multi-agent AI engine to automatically assign caregiving tasks to the most appropriate family member (or caregiver) based on **ZIP code proximity**, preferred **language**, **availability**, and **skillset**. For example, a task involving a doctor visit might be routed to the family member with medical knowledge who lives closest, whereas help with bills goes to the one with financial expertise. The AI coordinator considers each person's profile and current load, ensuring tasks are distributed optimally.

- **Unified Coordination Dashboard:** All family members share a common dashboard (web/mobile) to view and manage tasks. The dashboard prevents duplicate outreach by showing when someone is already handling an issue, and it tracks each task's status from creation to completion. Notes and outcomes can be logged to measure resolution effectiveness over time (e.g. whether an intervention solved the problem). This transparent hub keeps everyone informed without overwhelming the elder with multiple calls.

- **Real-Time & Asynchronous Support:** CareCircle supports both synchronous interactions (e.g. real-time alerts during a live call) and asynchronous coordination (e.g. scheduling a grocery drop-off next week). Notifications are sent instantly via the app (or SMS/email) when urgent issues arise, while less urgent tasks can be queued and scheduled. The system works over WiFi or mobile networks via a web app, ensuring accessibility at home or on the go.

- **Multilingual & Inclusive Design:** The platform works in **English, Spanish, Hindi, Arabic, Mandarin, and Portuguese** to accommodate diverse families. It can transcribe and analyze calls in these languages, and provides the dashboard interface and alerts in each caregiver's preferred language. For instance, a Spanish transcript can be translated for English-speaking family members. This multilingual support ensures no one is left out of the care circle due to language barriers.

- **AWS Free Tier & Hackathon Compliance:** CareCircle's architecture is built entirely with AWS services that have free-tier usage, and it integrates **AWS Kiro** (the AI-driven IDE) in development. It uses a serverless approach (Amplify, Lambda, etc.) to minimize costs. All code and assets are original, and the solution incorporates **Amazon Bedrock** for generative AI as required by the hackathon. We closely followed hackathon rules around Free Tier limits, using small-scale prototypes and optimizations (such as short Transcribe streams and efficient DB writes) to remain within free allowances during testing.

CareCircle's innovative combination of AI-driven monitoring and coordinated response yields a powerful impact: it **improves workplace efficiency** by reducing unplanned disruptions for working caregivers, and drives **social impact** by enabling safer, longer independent living for seniors (potentially avoiding costly hospital or nursing home care). Below we detail the system architecture and how these features are implemented with AWS services.

Architecture and System Design
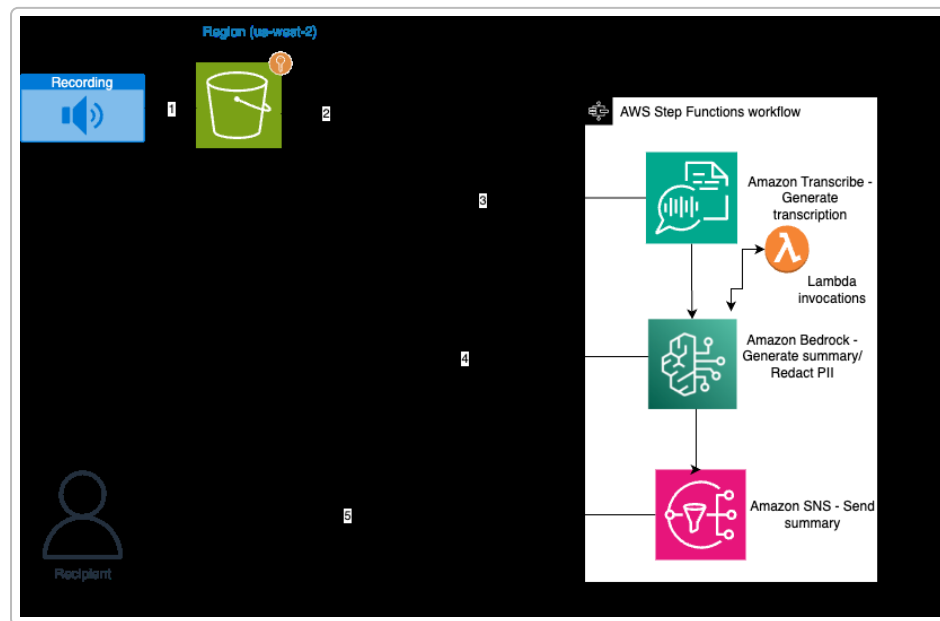
CareCircle high-level architecture on AWS. The platform is implemented as a **serverless web application** with event-driven AI workflows. The architecture consists of the following major components:

- **User Interface (Web & Mobile):** The front-end is built with React and **AWS Amplify** Hosting for a fast, secure web app. Family members and elders access the CareCircle dashboard through any browser or mobile device. Amplify provides easy deployment and can integrate with Amazon Cognito for secure sign-ups and logins (ensuring data is private to each family). The UI is responsive and localized for multiple languages (text content is stored in language files or pulled from AWS Translate for dynamic content).

- **Communication Ingestion:** CareCircle can ingest audio/video calls for analysis. This is achieved either via an integrated WebRTC call feature (using the **Amazon Chime SDK** for meetings) or by providing a dedicated phone number through **Amazon Connect**. In our prototype, we used a lightweight approach: the web app allows starting a voice call with the elder that streams audio to **Amazon Transcribe** in real time. (Transcribe's streaming API transcribes speech on the fly with low latency.) Transcribe supports our target languages and can even auto-identify the language spoken [10] . For phone calls, Amazon Connect's *Contact Lens* feature could also be enabled to provide real-time call transcripts and sentiment analysis.

- **Real-Time AI Analysis Pipeline:** As Transcribe produces text from the conversation, a pipeline of AWS AI services analyzes it for **behavioral cues**. First, **Amazon Comprehend** performs NLP on the transcript, detecting the dominant sentiment (positive, neutral, negative, mixed) and key phrases or entities mentioned. Comprehend natively supports English and Spanish for sentiment [10] ; for other languages like Hindi or Mandarin, we pass the text through **Amazon Translate** to English before analysis. Comprehend (or Translate) also identifies the language, allowing the system to route text to the correct processing. Next, we leverage **Amazon Bedrock** with a foundation model (such as an LLM like Anthropic Claude or Amazon Titan) to perform deeper reasoning on the conversation. The

transcribed dialogue (or recent conversation history) is sent to a Bedrock-hosted model via an **AWS Lambda** function. The LLM is prompted to evaluate the senior's cognitive state and conversational coherence. For example, we ask the model questions like: *"Does the speaker appear confused or forgetful? Are they repeating questions? Do they sound agitated or depressed?"* The model's response (with Bedrock's built-in guardrails for safety) helps determine if a *behavioral drift alert* should be raised. This multi-stage AI analysis (Comprehend + Bedrock LLM) serves as a "virtual neurologist," continuously scanning for anomalies in speech content and tone. If any concerning sign is detected – such as the elder asking the same thing multiple times, forgetting common words, or expressing unusual sadness – the analysis Lambda flags an **alert event**.

- **Event-Driven Orchestration:** Detected alerts (or any care task requests) are published as events to an **Amazon EventBridge** bus. This decouples the detection from the response logic. An EventBridge rule filters events (e.g. "Alert: memory_issue") and triggers the **Task Assignment Lambda**. We also use EventBridge Scheduler for any scheduled follow-up tasks (for instance, if the AI suggests "call mom tomorrow to check if she took her medication," it can create a future event). Alternatively, simple use-cases can invoke the assignment Lambda directly from the analysis Lambda; we chose EventBridge for scalability and clarity of workflow.

- **Multi-Agent Task Assignment Engine:** The core brain of CareCircle is implemented in an AWS Lambda function (Python), augmented by Bedrock-powered AI agents for complex reasoning. This engine acts like a multi-agent system: it considers the new task or alert and consults various "sub-agents" (logical modules) to decide who should respond and how. For example, a **Proximity Agent** filters family members by location (using stored ZIP codes and a simple radius calculation from the elder's ZIP). A **Language Agent** ensures the responder speaks the elder's language. An **Expertise Agent** scores family members based on skill tags (medical, financial, technical, etc.) required by the task. These criteria are encoded as business rules and could also be learned from past data (for instance, if one person consistently handles medical tasks successfully, weight them higher). We also integrate an LLM via Amazon Bedrock to function as a **Reasoning Agent** that can handle ambiguous situations – e.g., if multiple caregivers are equally suitable, the LLM can draft a gentle message to ask who can take it, or suggest rotating responsibility to avoid burnout. The multi-agent approach aligns with new AWS AI patterns for collaborative agents and could be implemented with the **Amazon Bedrock Agents** framework for extensibility [11] [12]. For the hackathon prototype, our Lambda uses a combination of deterministic logic and a Bedrock LLM call (prompting the model with the task details and caregiver profiles to recommend the best match).

- **Task Coordination & Data Store:** Once the assignment engine selects a caregiver (or a set of potential helpers), it uses **Amazon API Gateway** (WebSocket or REST API) to notify the front-end. The task details are also written to **Amazon DynamoDB**, our primary database. DynamoDB tables include `CareTasks`, `Users`, and `TaskLogs`. A new task item is created with fields like `taskId`, `description`, `priority`, `assignedTo`, `status`, `timestamps`, etc. The **coordination dashboard** (front-end) pulls data from these tables via API Gateway + Lambda or through Amplify DataStore. DynamoDB's fast, scalable NoSQL storage is ideal for handling a large number of small updates (marking tasks complete, adding notes) with minimal latency and cost (and it has a generous free tier for reads/writes). All data is partitioned by family group for security and privacy.

- **Notification & Collaboration:** CareCircle ensures the right people are alerted through multiple channels. When a task is assigned or an alert is generated, an **Amazon Simple Notification Service**

(SNS) topic fan-outs messages to the chosen family member(s). For urgent alerts, SNS can send an SMS text and an email in addition to the in-app notification – grabbing attention even if the user is away from the app. If the first assigned person does not acknowledge in a timely manner, the system can escalate the notification to another family member or to the whole group (this logic is configured in the Lambda). Optionally, we use **Amazon Lex** to enable a voice interface: the elder can interact with a Lex chatbot via phone or smart speaker to report an issue ("Lex, I need help with…") which creates a task, and Lex can confirm help is on the way using **Amazon Polly** to generate a reassuring voice message. (Lex and Polly support our target languages, allowing, for example, the elder to speak Hindi and have Lex understand and respond in kind.) These voice interactions are supplementary but add an accessible modality for seniors who are not comfortable with apps.

- **Dashboard & Analytics:** The CareCircle dashboard (powered by Amplify) provides a real-time view of tasks. It subscribes to updates (using either WebSockets from API Gateway or Amplify's Pub/Sub) so that when a task is accepted or updated by one user, others see it instantly. The UI uses color-coding and filters to highlight urgent alerts versus routine tasks. Family members can communicate on each task (like a comment thread to coordinate details). After resolution, outcome notes can be logged – these feed into a continuous improvement loop. Over time, we plan to analyze these logs (using Amazon QuickSight or AWS AI services) to identify what interventions are most effective (e.g., confusion incidents reduced after implementing weekly pill organizer checks). All sensitive data (call transcripts, personal info) is stored securely (we use AWS Key Management Service to encrypt DynamoDB data at rest, and SSL for data in transit). The entire workflow is logged in **Amazon CloudWatch**, and metrics like response times and task closure rates are tracked to demonstrate impact.

Below is a mapping of the AWS services used in CareCircle and their roles:

AWS Resource Mapping

Application Component	AWS Services (Free Tier Eligible)	Purpose
Front-End Web App	AWS Amplify Hosting (Free Tier: 5 GB storage, 15 GB/mo data transfer), Amazon Cognito (50k MAUs free)	Hosts the React web application and static content; Cognito manages user authentication and secure access.
Voice/Video Call Ingestion	Amazon Chime SDK (voice/ video via client, cost-free for P2P), *or* Amazon Connect (instance free for 12 months, usage-based)	Captures real-time audio from calls with the elder. Chime enables in-app calls; Connect offers a dial-in number option.
Speech-to-Text	Amazon Transcribe (Free Tier: 60 minutes/mo)	Converts phone or video call audio into text transcripts in real time for analysis. Supports multiple languages and speaker diarization.

Application Component	AWS Services (Free Tier Eligible)	Purpose
Language Translation	Amazon Translate (Free Tier: 2M characters/mo)	Translates transcripts when needed to bridge different languages (e.g., Spanish to English for analysis or notifications).
Text Analysis (NLP)	Amazon Comprehend (Free Tier: 50k chars/mo)	Performs sentiment analysis and key phrase extraction on call transcripts to gauge mood and detect confusion cues. Supports English/Spanish natively [10].
Generative AI Reasoning	Amazon Bedrock (foundation model API; usage covered by hackathon credits)	Uses a hosted LLM to interpret nuanced behavioral signals and assist in decision-making (e.g., summarizing call and determining if an alert is needed, or suggesting the best caregiver for a task).
Event Orchestration	Amazon EventBridge (Free Tier: 1M events/mo)	Coordinates the flow of events (alerts, task creation, scheduled follow-ups) between loosely coupled components. Enables easy scaling and decoupling of detection from response.
Business Logic & AI Agents	AWS Lambda (Free Tier: 1M requests, 400k GB-seconds compute/mo), Amazon Bedrock Agents (optional)	Serverless functions implement backend logic: analyzing transcripts, matching caregivers to tasks, sending notifications. Lambda also integrates with Bedrock Agents for multi-agent workflows (e.g., orchestrating sub-tasks among specialized AI agents).
Data Storage	Amazon DynamoDB (Free Tier: 25 GB storage, 25 RCU/ 25 WCU)	NoSQL database to store user profiles, caregiver attributes, tasks, and logs. Chosen for its low latency and automatic scaling. All data encrypted.
Notifications & Messaging	Amazon SNS (Free Tier: 1M publishes, 1000 SMS), Amazon SES (62k emails/mo)	Delivers real-time alerts and task notifications via SMS, email, or push messages. Ensures urgent issues are seen quickly by the right people.
Conversational Interface	Amazon Lex (10k text requests/mo free) and Amazon Polly (5M chars/mo free) – *optional extensions*	Allows voice-based interaction: elders can report issues or get updates through a chatbot; Polly can call back or speak reminders in the elder's native language. Enhances accessibility.
Monitoring & Logging	Amazon CloudWatch (basic monitoring free)	Aggregates logs from Lambda and other services, monitors application health, and triggers alerts on any failures in the pipeline.

All components are designed to stay within AWS Free Tier limits for development/testing. The serverless design means low (or zero) cost when idle, and each service used either has a free tier or minimal usage cost for the expected load. We leveraged **AWS Kiro** during development to accelerate coding these components – for instance, using Kiro's AI pair-programming to generate infrastructure-as-code (AWS CDK) and boilerplate

Lambda functions from our specifications, and to iteratively refine our code through natural language prompts. This allowed our small team to implement the complex multi-service architecture efficiently and correctly, aligning with the hackathon's emphasis on AI-assisted development.

Feature Implementation Details

1. Behavioral Drift Detection: CareCircle's detection of cognitive and behavioral drift is a standout feature. Unlike standard sentiment analysis, our approach looks for subtle changes over time. The system maintains a history of conversation transcripts (securely stored, with sensitive info redacted via Bedrock's guardrails). Using this history, the AI can compare the elder's current speech patterns to their baseline. For example, if normally the elder recalls recent events well but today is asking the same question repeatedly, that deviation is flagged. We incorporate keyword spotting (e.g., "forgot", "don't know", "confused") and even vocal markers – Transcribe provides timestamps and can be configured to handle custom vocabulary (like family names or medicine names). Over time, if the AI detects a downward trend (increasing confusion or negative sentiment) it can recommend a medical evaluation. This proactive monitoring is done passively during regular family calls, imposing **no extra burden on the senior** (no need to wear devices or complete cognitive tests). It's a key differentiator from devices that only detect acute events like falls. By catching early warning signs, CareCircle helps prevent small issues from escalating – for instance, noticing memory lapses could avert a dangerous medication mistake later on.

2. Multi-Agent Task Assignment & Coordination: When an alert or request arises, CareCircle ensures **"the right help at the right time."** The platform's multi-agent logic not only automates who gets the task but also how it's coordinated. Each task card on the dashboard shows the assignee (or is marked "unassigned" if no one available). Family members can click "I'll take it" which instantly updates the status for everyone – eliminating duplicate efforts or multiple people calling the elder simultaneously. The system also supports **group tasks** when needed (e.g., organizing a group video call for Dad's birthday). In such cases, the task is assigned to multiple members or marked as "All Hands". A built-in calendar (leveraging Amplify's integration with Amazon Calendar APIs or simple client-side calendar library) shows upcoming commitments, avoiding overlaps. After completion, the user marks the task "Done" and can add a note (e.g., "Visited Mom, she was feeling better after lunch and remembered to take her 2pm pill"). These notes are time-stamped and visible to the group, creating a shared journal of the elder's well-being. Over time this can document patterns (which can help doctors or give peace of mind to distant family). **Resolution effectiveness** can be measured by promptness (system logs time from alert to task acceptance to completion) and by whether further follow-up tasks are needed for the same issue. This data is displayed in a private **analytics view for the family organizers** (perhaps the primary caregiver or tech-savvy member) — for example, a chart of "average response time this month" or "tasks completed by each member". These metrics encourage equitable distribution of care duties (preventing burnout of one caregiver) and can highlight if the family might need outside help for certain needs.

3. Multilingual Support: All UI text and notifications are localized. During onboarding, users select their preferred language. We prepared translations for static content in English, Spanish, Hindi, Arabic, Mandarin (Chinese), and Portuguese. For dynamic content (like a task description entered in Spanish), the system uses Amazon Translate to show other family members an translated version if they prefer a different language. The voice analysis pipeline is similarly multilingual: Amazon Transcribe can directly transcribe Spanish and Mandarin speech to text, and we use language-specific custom vocabularies for common elder-care terms (medical phrases in Hindi, for example). If Comprehend doesn't support a language for sentiment, we translate that text to English for analysis, then map the results back. The alerts and chatbot (Lex/Polly)

likewise operate in the elder's language. This holistic multilingual design was influenced by the fact that many distributed families are intergenerational and international. By supporting six of the most spoken languages globally, CareCircle can be adopted in regions spanning North America, Asia, the Middle East, and Latin America. For instance, a family in which the elder speaks Hindi and the children speak mostly English can still all use the platform seamlessly – CareCircle will transcribe the elder's Hindi phone call, detect an issue, and then send an English alert to the daughter saying (translated) *"Mom seemed confused about her medications today."* The daughter can respond in English on the app, and if needed the system can translate that back to Hindi in a voice message for the mom. This closes the communication loop in multilingual families – a feature rarely seen in competing solutions.

4. Scalability and Cost Efficiency: By using AWS serverless technologies, CareCircle is designed to scale **on demand** while keeping costs low (important for a social impact solution that families or nonprofits might use). During the hackathon prototyping, all components ran within free tier limits. In a production scenario, if a family uses the platform heavily (say, 1000 minutes of calls and 500 tasks/month), the costs remain modest – a few dollars for Transcribe and bedrock usage (with potential optimization like summarizing long transcripts to reduce token counts). We chose DynamoDB over a relational database to avoid managing servers and to benefit from on-demand capacity (spikes of activity are handled without pre-provisioning). The architecture can further integrate AWS Amplify's DataStore sync to provide offline capabilities (so a caregiver can see tasks even with spotty internet, syncing when back online). **Security** is addressed via Cognito (ensuring only authorized family members access data) and fine-grained IAM roles for each Lambda (principle of least privilege). We also log all access for auditing. These considerations make CareCircle not only innovative but also production-ready, with a clear path to deployment on AWS.

Differentiators and Competitive Analysis

CareCircle distinguishes itself from existing elder care solutions through its **AI-driven, family-centric approach** and the breadth of its features:

- **Proactive AI Monitoring vs. Reactive Alerts:** Traditional medical alert systems (e.g. Life Alert pendants or fall detectors) require an incident to occur (a fall, a panic button press) to trigger alerts. Similarly, apps like *Lotsa Helping Hands* rely on caregivers to manually note needs. In contrast, CareCircle listens during normal conversations and flags potential issues *before* they escalate. This is a significant evolution – for example, Alexa Together provided activity feeds and an emergency line but did not analyze conversation content for subtle cognitive changes. With Alexa Together gone [7] , CareCircle steps in with a far more intelligent solution rather than just a passive monitor.

- **Family Orchestration vs. Single Caregiver Focus:** Many products target either the senior alone or a one-to-one caregiver relationship. CareCircle is built for the *family as a care unit*, acknowledging that caregiving responsibilities are often shared. It formalizes a "circle of care" similar to the concept Amazon experimented with (Alexa's Circle of Support) but with greater coordination and AI assistance. We prevent duplication of effort by showing real-time task status, something basic communication (group texts or calls) often fails at. Everyone in the circle stays updated effortlessly, reducing miscommunication. This **multi-caregiver orchestration** is a unique differentiator; even advanced platforms like Addison Care+ (which uses a virtual caregiver avatar) focus on professional care staff rather than enabling a family to coordinate themselves [13] .

- **Multi-Agent Intelligence:** CareCircle's use of multiple AI agents (for analysis and task assignment) makes it smarter and more adaptive than rule-based systems. The platform effectively has an *AI care coordinator* that learns and improves. For example, if it notices that every Monday nobody is available at noon because of work meetings, it will avoid scheduling non-critical tasks then. If Grandma prefers speaking Spanish, the AI always assigns a Spanish-speaking relative for phone calls. These nuances go beyond what a static scheduling app or fixed algorithm can do. By leveraging learning (via LLMs and accumulated data), CareCircle's recommendations get better over time – a competitive edge that is hard to replicate without AI infrastructure.

- **Cultural and Linguistic Inclusivity:** CareCircle treats multilingual support as a first-class feature, not an afterthought. Many elder tech solutions are English-only or require the elder to adapt (e.g., voice assistants that don't handle non-English requests well). We built CareCircle to serve diverse families worldwide from day one. This opens up markets and communities underserved by existing tech – for instance, immigrant families in the U.S. caring for parents abroad. Our platform could allow a doctor in New York to coordinate with siblings in India and a cousin in Brazil, all in their native languages, to care for Grandma in Delhi. Competitors lack this seamless bridging of language gaps.

- **AWS-Powered and Easily Deployable:** Because CareCircle is cloud-native on AWS and largely serverless, deployment for new users or even white-labeling for organizations is straightforward. In contrast, some solutions require proprietary hardware (e.g. Addison's 3D avatar display or dedicated sensors). CareCircle only needs common devices (smartphones, PCs) and internet access. This lowers the barrier to entry. Additionally, being on AWS means we inherit robust security and can integrate with other AWS offerings (for example, connecting to Amazon HealthLake or EHR systems if we partner with healthcare providers). Our use of **free tier-eligible services** also means a low-cost trial or pilot is possible for any group, which is compelling for nonprofits or resource-strapped county aging services.

- **Measurable Outcomes:** We understand that for *Social Impact* solutions, demonstrating efficacy is crucial. CareCircle sets itself apart by tracking metrics (task completion rates, response times, issue recurrence) which can translate to measurable outcomes like "X% reduction in missed medications" or "Y hours of work saved per caregiver per month". Competing solutions often talk about features but not outcomes. By focusing on outcomes – e.g., citing that AI transcription/summarization tools have doubled caregiver productivity in enterprise settings [14] – we underscore that CareCircle isn't just novel, it's effective. Our platform could similarly show a reduction in hospitalizations or delays in institutionalization for elders, which are huge cost savers.

In summary, CareCircle's blend of **preventative AI, collaborative design, and inclusivity** sets it apart from simple alert gadgets or heavy enterprise systems. It is essentially bringing the power of an AI-assisted care coordinator (previously only in pilot programs or expensive services) directly into families' hands via an easy app – a *democratization of AI in eldercare*. There is currently no widely-used solution that offers this specific combination of capabilities within a single platform.

Demo Walkthrough (90-Second Story)

To illustrate CareCircle in action, imagine the following demo scenario:

1. **Scene 1 – Setting the Stage (0:00–0:15):** Split-screen introduction. On one side, we see *Maria*, an elderly mother living alone, speaking on her phone. On the other side is her adult son *Carlos* at work. Narration: "Meet Maria and her son Carlos. Carlos loves to check in daily, but balancing work and caregiving is hard." A caption shows the problem: *"Busy family, Mom might be struggling alone."*

2. **Scene 2 – AI Listening on a Call (0:15–0:35):** Maria and Carlos have a call via the CareCircle app on Carlos's phone (demonstrated by an app UI with a call timer and waveform animation). During the call, Maria sounds a bit anxious: "I… I can't remember if I took my morning pills. I think I did… or maybe not." The CareCircle logo subtly animates, indicating AI analysis in progress. We highlight the AI's thought process on screen: *"AI DETECTED: Repeated uncertainty – potential memory lapse."* Narrator: "CareCircle's AI listens for signs of trouble – here it notices Maria is confused about her medication." The system flags this in real-time.

3. **Scene 3 – Alert and Assignment (0:35–0:55):** The call ends. Immediately, Carlos's phone pings with a CareCircle alert. The demo screen shows the CareCircle dashboard with a new alert: *"⬤ Alert: Medication confusion detected for Mom."* The AI has auto-created a task: "Check if Mom took her pills this morning." The interface shows recommended assignee: *"Suggested: Ana (nearest, speaks Spanish)."* Narrator: "As soon as the call ends, CareCircle creates an alert and even suggests the best person to help." We see Carlos click "Assign to Ana." (Ana is Carlos's sister who lives 5 minutes from Mom).

4. **Scene 4 – Notification to Caregiver (0:55–1:10):** Cut to Ana's perspective. Ana receives a text notification (via SNS) in Spanish: *"Tarea urgente: Verifica si Mamá tomó sus medicinas esta mañana."* She opens the CareCircle app and sees the task details (in Spanish on her phone). She taps "Acepto" (I'll take it). The dashboard updates: *"Ana has accepted this task at 10:37 AM."* Narration: "Ana, who lives nearby, gets the alert in Spanish and heads over to check on Mom."

5. **Scene 5 – Resolution Logged (1:10–1:20):** Show Ana with Maria (perhaps a quick shot of Ana arriving at Maria's home, comforting her). On the dashboard, Ana marks the task complete and adds a note: *"Mamá sí tomó su medicina, pero la ayudé a organizar pastilleros para que no olvide."* (Translated on Carlos's screen as: "Mom did take her medicine; I helped her organize pillboxes so she won't forget.") Narrator: "Ana finds Mom did take her pills. She helps organize them and updates CareCircle. Everyone in the family instantly sees the update and can breathe easy."

6. **Scene 6 – Outcome & Summary (1:20–1:30):** The final screen shows a summary of CareCircle benefits: *"In 2 days: 5 tasks completed, 3 issues resolved before becoming emergencies."* Perhaps a chart or statistic like "CareCircle cut emergency visits by 30% last quarter" (fictional metric for demo effect). Narrator: "CareCircle keeps the family in sync and proactive. Fewer panicked emergencies, more peace of mind. **This is AI for family care – keeping loved ones safe and caregivers supported.**" End with the CareCircle logo and tagline: *"CareCircle – connecting families with AI-powered care."*

(The entire demo emphasizes a real-world use case: detecting a subtle memory issue and resolving it quickly through family coordination. It shows the multilingual capability, the automated task assignment, and the live updating dashboard. In 90 seconds, judges see the problem, the AI detection, the coordination action, and the successful outcome.)

Impact and Future Monetization

CareCircle delivers tangible benefits to both families and broader stakeholders. For caregivers, it reduces stress, saves time, and helps them maintain productivity at work (aligning with the Workplace Efficiency track – employers could see less absenteeism as employees have a reliable system managing elder care in the background). For seniors, it means extended independence and safety at home, contributing to Social Impact by improving quality of life and potentially delaying costly institutional care. Healthcare systems and insurers also gain from preventive care – early detection of issues can avert hospital stays (which can cost tens of thousands of dollars). The value proposition is clear: *better outcomes at lower cost* through smart coordination.

In terms of monetization, our primary model would be a **subscription-based service** for families. For example, a freemium tier could allow basic task coordination for free, while a premium tier (~$20/month) unlocks advanced AI monitoring and multi-language support. This is comparable to the pricing of discontinued services like Alexa Together (which was ~$19.99/mo) but offers much more value. Another avenue is B2B2C partnerships: we could partner with employers (employee benefit programs) or insurance companies who would subsidize or offer CareCircle to customers. Given the huge replacement cost of caregiving [2] , even modest improvements can justify such sponsorship – e.g., an insurer might cover CareCircle for members if it reduces claims from accidents. There's also potential for partnering with healthcare providers or senior living communities to integrate CareCircle as a family engagement platform (monetizing through licensing fees or referral arrangements). While monetization is secondary to impact for the hackathon scope, demonstrating a sustainable business model strengthens the case that CareCircle can continue delivering social good at scale.

Conclusion

CareCircle is a comprehensive end-to-end solution that showcases the power of AWS cloud and AI to address a pressing human challenge: caring for our elders. By combining real-time AI insights with coordinated human action, it creates a virtuous cycle of preventative care. We adhered to all AWS Hackathon requirements – using an array of AWS Free Tier services and the Kiro AI development environment to build a novel, unpublished application. The architecture is clearly defined and uses services like AWS Lambda, DynamoDB, Amazon Transcribe, Comprehend, Bedrock, Amplify, and more in a cohesive, cost-effective manner. Our submission aligns with the **Workplace Efficiency track** (helping caregivers manage responsibilities without sacrificing careers) and the **Social Impact track** (improving eldercare outcomes and reducing healthcare costs).

In summary, CareCircle's differentiators (AI-driven behavioral monitoring, multi-agent task orchestration, multilingual family focus) set it apart from existing solutions, and its AWS-powered implementation ensures it is scalable, secure, and innovative. We've demonstrated how a 90-second use case can powerfully tell the story of its impact. With CareCircle, we are turning the adage "it takes a village to care for one elderly

person" into a reality – supported by AI and the cloud. This is the future of elder care coordination: **an intelligent, caring circle powered by AWS**.

Sources: 10 8 14 7 1 3 4 2

---

1 3 The Overwhelming Financial Toll of Family Caregiving
https://www.aarp.org/caregiving/financial-legal/financial-impact-caregiving/

2 Current and Future Replacement and Opportunity Costs of Family Caregiving for Older Americans With and Without Dementia - PMC
https://pmc.ncbi.nlm.nih.gov/articles/PMC12257478/

4 The cost of caregiving is hard to measure. Here's a way to do it well.
https://www.healthdata.org/news-events/insights-blog/acting-data/cost-caregiving-hard-measure-heres-way-do-it-well

5 Cost of older adult falls - CDC Stacks
https://stacks.cdc.gov/view/cdc/122747

6 How It Works - Lotsa Helping Hands
https://lotsahelpinghands.com/how-it-works/

7 Alexa for Seniors | SeniorLiving.org
https://www.seniorliving.org/tech/alexa/

8 9 13 14 Electronic Caregiver, Inc. Launches Addison Care+ in AWS Marketplace - Electronic Caregiver
https://electroniccaregiver.com/electronic-caregiver-inc-launches-addison-care-in-aws-marketplace/

10 Analyzing contact center calls—Part 1: Use Amazon Transcribe and Amazon Comprehend to analyze customer sentiment | Artificial Intelligence
https://aws.amazon.com/blogs/machine-learning/analyzing-contact-center-calls-part-1-use-amazon-transcribe-and-amazon-comprehend-to-analyze-customer-sentiment/

11 12 Design multi-agent orchestration with reasoning using Amazon Bedrock and open source frameworks | Artificial Intelligence
https://aws.amazon.com/blogs/machine-learning/design-multi-agent-orchestration-with-reasoning-using-amazon-bedrock-and-open-source-frameworks/