SSCP HACKATHON · GenAI.mil TRACK

TravelClaim AI

Automated Travel Reimbursement for the Rank and File

Problem The DD Form 1351-2 (Travel Voucher) is one of the most filled and most dreaded forms in the US military. Between the complexity of the Joint Travel Regulations, the Defense Travel System, per diem rules, GTCC eligibility, and multi-branch variation — a simple TDY trip can take hours of administrative work and weeks of waiting for reimbursement.

Solution TravelClaim AI is an offline-capable, AI-powered assistant that lets a soldier describe their travel in plain English, answers their policy questions with citations to the JTR, fills the DD 1351-2 automatically, and validates uploaded receipts — cutting processing time from days to minutes.

822	9,754	1,241	18.2%
Reddit posts found	Total upvotes	Peak post score	Complaint rate

Reddit analysis across r/army, r/AirForce, r/navy, r/Veterans, r/nationalguard — 2012 to 2026 · Data source: Arctic Shift

1. The Problem — Validated by Soldiers Themselves

Travel reimbursement is one of the highest-frequency administrative burdens across all military branches. It affects every service member who travels for training, PCS moves, or TDY assignments — which is effectively the entire force.

What the data shows

- 822 posts across 5 subreddits spanning 14 years (2012–2026) about travel voucher and reimbursement pain
- Volume grew 136× from 2012 to a sustained peak of 128–136 posts/year in 2022–2024, tracking DTS expansion
- 9,754 total upvotes — soldiers recognize and validate each other's frustration at scale
- 18.2% of posts contain 2 or more active frustration signals ("still waiting", "rejected", "broken", "months")
- Top post: "Per Diem should be 100% on travel days" — 1,241 upvotes
- "Just want paid out for my travel voucher" — 1,032 upvotes. That title is a product brief.

> **Key insight from the data**
>
> "months" appeared in 97 posts, "weeks" in 71 — soldiers are waiting months for money they already spent out of pocket.
>
> "DTS" (the Defense Travel System) appears 158 times — the tool meant to help is itself a named pain point.
>
> The 29 direct mentions of "1351" in titles are the tip of the iceberg; most soldiers don't know the form number.

Why this matters operationally

- Every hour spent on travel paperwork is an hour not spent on mission readiness
- Finance offices are understaffed, especially post-DOGE workforce reductions — backlogs are growing
- Junior soldiers (E-1 to E-5) bear the highest burden and have the least institutional knowledge to navigate JTR
- PCS season (summer) and post-exercise periods create surge demand that overwhelms manual processes

2. TravelClaim AI — The Solution

2.1 Core Concept

A soldier opens TravelClaim AI on any device (phone, tablet, laptop — or offline). They describe their travel situation in plain English. The AI answers their policy questions with citations to the JTR, fills the DD 1351-2 PDF automatically, and validates their receipts. The Finance NCO reviews a pre-filled, validated package instead of a blank form.

2.2 The Three Capabilities

Capability 1 — JTR Policy Assistant (RAG chatbot)

A retrieval-augmented generation (RAG) system trained on the Joint Travel Regulations, JFTR, and relevant Army Regulations. Soldiers can ask natural language questions and receive accurate, cited answers.

- "Am I entitled to per diem on travel days?" → cites JTR par. 0503
- "What receipts do I need for my lodging claim?" → lists required documentation
- "Can I use my GTCC for this expense?" → validates against GTCC policy
- "My TDY was extended 3 days — does my authorization need to change?" → step-by-step guidance

Capability 2 — Automatic Form Fill (DD 1351-2)

Soldier inputs travel details conversationally. The AI extracts all required fields and populates the DD Form 1351-2 PDF. The output is a download-ready, correctly filled form.

- Natural language: "I drove from Fort Campbell to Fort Bragg, 847 miles, stayed 4 nights at the Holiday Inn ($129/night), TDY dates June 3–9"
- AI parses: origin, destination, mileage, lodging cost per night, dates, per diem entitlements
- Output: completed DD 1351-2 PDF with all fields populated, ready to sign and submit
- Validates against JTR rules before output (lodging caps, meal deductions, mileage rates)

Capability 3 — Receipt Intelligence (Phase 3)

Soldier photographs receipts with their phone. AI extracts line items, validates each against JTR reimbursable categories, flags GTCC-eligible vs out-of-pocket expenses, and adds to the claim automatically.

- Vision model reads receipt: hotel folio, Uber receipt, meal receipt, parking ticket
- Validates: is this expense category reimbursable under JTR?
- Flags: should this go on GTCC or personal card for reimbursement?
- Auto-populates the relevant line items in the DD 1351-2

3. Build Plan — Hackathon Execution

3.1 Phase Overview

The build is structured in four phases, each with a concrete deliverable. Phases 1–2 are the core demo; Phases 3–4 are stretch goals if time allows.

Phase	What we build	Deliverable	Timeline	Owner
1 — RAG Core	Ingest JTR + JFTR into vector DB. Build chatbot that answers travel policy questions with citations.	Working RAG chatbot (local)	Hours 0–6	All 3 devs
2 — Form Fill	NL input → parse fields → auto-populate DD 1351-2 PDF. Army first, then generalize.	Filled PDF download	Hours 6–14	Dev 1 + 2
3 — Receipt AI	Upload receipt image → GPT Vision extracts fields → validates vs JTR rules → flags GTCC eligibility.	Receipt → form pipeline	Hours 14–22	Dev 2 + 3
4 — Frontend	Next.js UI with role-based views (Soldier, Finance NCO, Unit Commander). Supabase auth.	Live demo URL	Hours 20–28	Dev 1 + 3

3.2 Phase 1 — RAG Pipeline (Hours 0–6)

Step 1: Download the corpus

- Joint Travel Regulations (JTR): https://www.travel.dod.mil/Policy-And-Regulations/Joint-Travel-Regulations/
- JFTR (uniformed services supplement): download the bulk PDF
- Army Regulation 37-104-4 (military pay and allowances policy)
- GSA Per Diem rates: available via GSA Open API (no auth required)

Step 2: Chunk and embed

- Split PDFs into chunks of ~500 tokens with 50-token overlap
- Embed using OpenAI text-embedding-3-small ($0.02/1M tokens — the entire JTR is < 1M tokens)
- Store vectors in pgvector (Supabase) or ChromaDB locally
- Include metadata: source document, chapter, paragraph reference for citations

Step 3: Build the RAG chain

- User query → embed → similarity search → retrieve top 5 chunks → pass to Claude with system prompt
- System prompt instructs Claude to: only answer from retrieved context, always cite paragraph/section, flag if answer is uncertain
- Test with 10 common soldier questions before moving to Phase 2

Demo test questions for Phase 1 validation

1. What is the per diem rate for lodging in Washington DC?
2. Am I entitled to full per diem on the first and last day of travel?
3. What receipts are required for a lodging claim over $75?
4. Can I claim mileage if I drove my POV instead of flying?
5. What is the GTCC policy for rental cars?

3.3 Phase 2 — Form Fill Engine (Hours 6–14)

DD 1351-2 field mapping

- Download the official fillable PDF from the DoD forms library
- Map all 40+ form fields to their JTR data sources
- Build the extraction prompt: given a travel description, output a JSON object with all form fields
- Use pdf-lib (JavaScript) to write extracted values into the PDF fields programmatically
- Output: a pre-filled PDF the soldier can download, sign, and submit

Validation layer

- Before filling the form, run a validation pass: are lodging costs within the per diem cap?
- Flag any line items that need Finance review (excess lodging, missing receipts)
- Output a plain-language summary: "Your claim totals $847. The lodging on Night 3 exceeds the GSA cap by $22 — you'll need a justification memo."

3.4 Phase 3 — Receipt Intelligence (Hours 14–22)

Vision pipeline

- User uploads receipt photo via the UI
- Claude Vision (claude-sonnet-4-20250514) extracts: vendor, date, total, line items, payment method
- Classification prompt: for each line item, determine JTR reimbursability category
- GTCC check: flag which expenses must go on the Government Travel Charge Card
- Output: structured JSON of validated, classified expenses ready to populate the form

3.5 Phase 4 — Frontend (Hours 20–28)

User roles

- Soldier view: chat interface + form fill + receipt upload
- Finance NCO view: review queue of submitted claims, validation flags highlighted
- Unit Commander view: team travel status dashboard, pending approvals

Tech approach

- Next.js 14 App Router + Tailwind CSS — deploy to Vercel in one push

- Supabase Auth for role-based access (soldier, NCO, commander)
- Supabase Postgres + pgvector for the vector store (no extra infra)
- API routes in Next.js call Anthropic API and handle PDF generation

4. Technology Stack

Chosen for team familiarity, hackathon speed, and offline viability.

Layer	Technology	Why
Frontend	Next.js 14 + Tailwind CSS	Team familiarity, fast dev, Vercel deploy in 1 click
Auth + DB	Supabase (Postgres + Auth)	Row-level security, real-time, team knows it
Vector Store	pgvector (on Supabase) or ChromaDB local	JTR chunks stored as embeddings; no extra infra
Embeddings	OpenAI text-embedding-3-small	Fast, cheap, high quality for regulatory text
LLM	Claude claude-sonnet-4-20250514 via Anthropic API	Best at following complex rules, long context for JTR
PDF Fill	pdf-lib (JS) or PyMuPDF (Python)	Fill DD 1351-2 fields programmatically
Vision / OCR	Claude Vision (receipt images)	Extract line items, dates, amounts from photos
Deploy	Vercel (frontend) + Railway or Render (API)	Zero-config, free tier covers hackathon demo

Offline-first strategy (required by the track)

Phase 1 demo: Ollama + llama3.1:8b-instruct-q4 running locally, ChromaDB on disk. No internet required.

Phase 4 demo: cloud-hosted for the live demo URL, but the same code runs offline with env var swap.

The JTR corpus is static — embed once, store locally, no re-fetching needed.

5. Team & Roles

3 Developers — Division of Work

Area	Dev 1	Dev 2	Dev 3
RAG pipeline	**Lead**	Support	
PDF form fill	**Lead**	Support	
Receipt vision		**Lead**	Support
Next.js frontend	Support		**Lead**
Supabase / DB		Support	**Lead**
Prompt engineering	**Lead**	Support	Support
Demo + pitch deck	Support		**Lead**

6. Pitch Narrative — What to Say to Judges

Opening (30 seconds)

"The DD Form 1351-2 is the most hated piece of paper in the US military. Every service member who travels has to fill it out. Most fill it out wrong. Finance sends it back. They wait weeks. We found 822 Reddit posts, 9,754 upvotes, and one post that just says 'Just want paid out for my travel voucher' — with over 1,000 upvotes. That's your user research. TravelClaim AI fixes this."

Problem (60 seconds)

The JTR is 900+ pages. DTS is broken and soldiers hate it. Finance offices are short-staffed. A junior soldier on their first TDY has no idea what they can claim, what receipts to keep, or how to fill the form. They lose money they're entitled to, or they fill it wrong and create a Finance backlog. This drains readiness.

Demo flow

- Show the Reddit analysis slide — 822 posts, 18.2% complaint rate, the trend chart
- Open TravelClaim AI — type: "I did a 5-day TDY to Fort Meade, drove my POV 340 miles each way, stayed at a hotel for $145/night, and had two meals per day on my GTCC"
- AI answers: am I entitled to per diem? With JTR citation
- AI fills the DD 1351-2 PDF — soldier downloads it
- Upload a hotel receipt photo — AI extracts fields and validates them
- Show the Finance NCO review queue — clean, flagged, ready to approve

Why we win

- Real data validation: we didn't guess this was a problem — we proved it with 822 posts
- Offline-first: works on a local laptop at a FOB with no internet
- End-to-end: not just a chatbot — it generates the actual form

- Generalizable: Army first, then Air Force, Navy, Marines — same JTR, same form
- Agentic: the AI doesn't just answer questions, it takes action (fills forms, validates receipts)

7. Beyond the Hackathon — What This Becomes

The hackathon build is a proof of concept. The full product vision scales to the entire DoD travel workflow.

- Multi-branch support: Air Force IMT 4009, Navy NAVCOMPT 3065 — same RAG, different forms
- DTS integration: generate a pre-filled DTS authorization from the same NL input
- Commanding officer approval workflow: digital routing with audit trail
- Real-time per diem lookup: GSA API integration for current city rates
- Audit-ready: every claim decision is logged with the JTR citation that justified it
- DOGE-adjacent: reduces Finance staffing requirements through automation — saves DoD money

> **The long-term pitch to DoD**
> Finance offices process millions of travel claims per year. If TravelClaim AI reduces handling time by even 30 minutes per claim, the math on 3 million service members traveling annually is significant. This is the kind of tool DOGE said it wanted to build — but didn't have time to finish.