Manual Testing Log
Cloud Cost Intelligence Platform

Testers: Michael Allen (PM/Tech Lead)

CMSC 495 Computer Science Capstone

February 13–15, 2026

Table of Contents

Overview

Manual testing was conducted across five sessions over three days as a complement to the 143 automated tests (pytest, Jest, Playwright). These sessions followed a structured 18-step demo scenario mapped to all 8 functional requirements, followed by targeted regression testing after defect fixes. Manual testing discovered 14 defects (DEF-008 through DEF-021), of which 13 were resolved during Phase II.

Environment

Component	Detail
OS	macOS (ARM M2)
Browser	Google Chrome
Docker	Docker 29.2.0
Frontend	nginx: 8080 serving static files
Backend	Flask: 5001 via Gunicorn
Database	Azure SQL Serverless Gen5 (40K+ usage records, 10 clients, 3 providers, 30 services)
Auth	Microsoft Entra ID device code flow
Network	nginx reverse proxy

Session 1: 18-Step Demo Scenario (Feb 13)

Purpose: End-to-end walkthrough of all functional requirements using live Azure SQL data. Docker containers running, Azure SQL authenticated via device code flow, 40K+ usage records loaded.

Scenario 1: IT Manager Reviews Monthly Costs

Step	Action	FR	Expected Result	Actual Result	Status
1	Open dashboard at localhost:8080	FR-01	Cost cards display AWS, Azure, GCP totals	Total Spend, provider cards render with live dollar amounts	PASS
2	Scroll to Resource Metrics section	FR-08	Utilization stats (avg/max/min) displayed	No DOM element exists for metrics section	FAIL — DEF-012
3	Select AWS from Provider filter, click Apply	FR-03	Dashboard filters to AWS-only costs	Provider dropdown used ID=1, database uses 2001. Filter returned no data	FAIL — DEF-013
4	View Spending Trends chart	FR-02	Multi-day trend line with daily cost data	Chart renders single data point (only 100 of 40K records loaded)	FAIL — DEF-010
5	Change date range to Last 30 Days	FR-02	Trend chart updates to 30-day view	Single point remains (same root cause as Step 4)	FAIL — DEF-010
6	Click Waste Alerts in nav	FR-04	Table of underutilized resources with costs	Cosmos, Elastic Beanstalk, Artifact Registry displayed with 0% util, daily cost, monthly savings	PASS
7	Click Settings, select client, enter $20K, Save	FR-07	Budget saves successfully	Client dropdown empty — could not select client	FAIL — DEF-008
8	Click Export CSV and Export PDF	FR-06	Files download with cost data	Both files download. CSV has 5 columns (limited). PDF functional	PASS

Scenario 2: DevOps Engineer Finds Unused Resources

Step	Action	FR	Expected Result	Actual Result	Status
9	Click Waste Alerts tab	FR-04	Resource list with utilization data	Table populated: service name, provider, utilization %, daily cost, potential savings	PASS
10	Click View Details on Artifact Registry	FR-04	Detail banner with savings summary	Banner: Artifact Registry — 0% utilization, save $46.74/month	PASS
11	Review waste alert severity indicators	FR-04	Critical/Warning/Info severity badges	Severity badges display correctly based on utilization thresholds	PASS
12	Review category breakdown bars	FR-04	Visual breakdown by service category	Category bars render with proportional cost distribution	PASS
13	Click Recommendations in nav	FR-05	Rightsizing suggestions with savings	Cards: Rightsize Cosmos ($162.07), Elastic Beanstalk ($12.16), Artifact Registry ($46.74)	PASS
14	Review recommendation detail accordion	FR-05	Plus/minus analysis per recommendation	Accordion expands with effort, savings rate, phased rollout plan	PASS

Scenario 3: Finance User Generates Monthly Report

Step	Action	FR	Expected Result	Actual Result	Status
15	Return to Dashboard	FR-01	Cost cards display correctly	Dashboard renders with correct totals ($4,699.73 confirmed)	PASS
16	Check Resource Metrics section	FR-08	Utilization stats visible	Still no UI element (same as Step 2)	FAIL — DEF-012
17	Change date range to Last 7 Days	FR-03	Dashboard updates with filtered data	$0.00 (correct — seed data >7 days old). Last 30 Days restores $380.08. Filter functional	PASS
18	Export PDF	FR-06	PDF downloads with cost summary	PDF downloads successfully	PASS

Session 1 Summary

FR	Requirement	Status	Blocking Defect
FR-01	View Cost Dashboard	PASS	—
FR-02	View Spending Trends	FAIL	DEF-010: Fetch limit too low
FR-03	Filter by Provider/Service	FAIL	DEF-013: Provider ID mismatch
FR-04	View Waste Alerts	PASS	—
FR-05	View Recommendations	PASS	—
FR-06	Export Reports	PASS	Format polish needed
FR-07	Set Budget Thresholds	FAIL	DEF-008: Client dropdown empty
FR-08	View Resource Metrics	FAIL	DEF-012: No DOM element

Result: 5 of 8 FRs demo-ready. 3 blocked by defects.

Session 2: Provider Filter Fix Verification (Feb 14)

Purpose: Verify DEF-013/DEF-014 fix — provider IDs corrected to 2001/2002/2003 in config.js. Branch: fix/def-014-provider-filter-ids.

Test ID	Action	Expected	Actual	Status
MT-201	Select AWS from Provider filter	Dashboard shows AWS-only costs	Console: Filtered by provider 2001: 393 records. AWS $309.09, Azure $0.00	PASS
MT-202	Check Service Type dropdown with AWS selected	Only AWS services shown	Elastic Beanstalk, EC2, Lambda, S3, DynamoDB, RDS, EFS, EKS, Fargate — all AWS	PASS
MT-203	Switch to Azure in Provider filter	Dashboard shows Azure-only costs	Azure costs display, AWS $0.00, service dropdown cascades to Azure services	PASS
MT-204	Switch to GCP in Provider filter	Dashboard shows GCP-only costs	GCP costs display, cascading filter shows GCP services	PASS
MT-205	Select specific service (Elastic Beanstalk)	Dashboard filters to single service	Correct service-level data displayed	PASS
MT-206	Select Lambda (no usage data)	Dashboard shows no data message	No data found banner — correct (no usage records)	PASS
MT-207	Verify service filter shows inactive services	Only services with usage data shown	All 30+ services listed including 27 with no usage data	FAIL — DEF-016

Defects Found: DEF-016 (Service filter shows services with no usage data)

Session 3: Budget Save & Export Verification (Feb 14)

Purpose: Test remaining unverified demo steps (7, 17, 18) after filter fixes. Branch: feature/export-reports.

Test ID	Action	Expected	Actual	Status
MT-301	Date range: select Last 7 Days	Trend chart and cards update	$0.00 displayed (correct — seed data outside 7-day window)	PASS
MT-302	Date range: select Last 90 Days	Full dataset shown	All data loads, chart expands	PASS
MT-303	Budget Save: select client, enter $20K, Save	Settings persist	Failed to save settings. Console: 405 Method Not Allowed — POST vs PATCH	FAIL — DEF-017
MT-304	PDF Export button	PDF downloads	PDF downloads with usage data for selected date range	PASS

Defects Found: DEF-017 (Budget save sends POST instead of PATCH → 405 error)

Session 4: Post-Fix Full Regression (Feb 15)

Purpose: Validate analysis.js integration, waste alerts rendering, recommendations, budget save after PATCH fix, and export filter sync. Branch: feature/export-reports (post analysis.js commit). Docker restarted, Azure SQL re-authenticated after idle timeout.

Dashboard & Navigation

Test ID	Action	Expected	Actual	Status
MT-401	Select Anderson Consulting from Client dropdown	Dashboard filters to Anderson data	Cost cards, trend chart update to Anderson-specific data	PASS
MT-402	Verify no stale data flash on Settings load	Settings form starts empty/loading	Placeholders show Loading... until API returns	PASS

Waste Alerts (FR-04)

Test ID	Action	Expected	Actual	Status
MT-403	Click Waste Alerts with Anderson selected	Summary banner and severity cards render	Money on the Table banner: total spend, savings, budget status, over-budget flag	PASS
MT-404	Review severity classification	Resources classified by utilization threshold	Critical (<20%), Warning (20–50%), Optimized (>50%) badges render correctly	PASS
MT-405	Verify waste detection logic	Underutilized resources flagged with savings	Cosmos: 0% util, $162.07/mo savings. Elastic Beanstalk: 0% util, $12.16/mo savings	PASS
MT-406	Verify trend calculation	30-day trend shows cost direction	Trend splits data at midpoint, compares recent vs prior period	PASS
MT-407	Review category breakdown bars	Visual cost distribution by category	Bars render proportional to category spend totals	PASS

| MT-408 | Verify budget status indicator | Over-budget clients flagged | Anderson shows over-budget (actual spend >> $1,000 budget) | PASS |

Recommendations (FR-05)

Test ID	Action	Expected	Actual	Status
MT-409	Click Recommendations tab	Savings roadmap with phased cards	Savings Roadmap banner with current → target → recoverable. Phase 1/2/3 cards render	PASS
MT-410	Expand recommendation accordion	Plus/minus analysis per recommendation	Each card: implementation effort (hours, cost, downtime), savings rate, rollout plan	PASS
MT-411	Verify savings calculation	Dollar amounts match template rates	Savings = monthly_cost × template.savingsRate. Reserved pricing 30%, rightsizing 20–25%	PASS
MT-412	Check client attribution on recommendations	Each recommendation shows owning client	No client name displayed — service name only	FAIL — DEF-020
MT-413	Check waste alerts client attribution	Waste alerts show correct client name	Multiple displayed instead of actual client name	FAIL — DEF-021

Budget Settings (FR-07)

Test ID	Action	Expected	Actual	Status
MT-414	Select client, view pre-populated budget	Budget loads from GET /budgets	Budget amount, limit, threshold display correctly	PASS
MT-415	Change budget to $20,000, click Save	PATCH /budgets/{id} succeeds	Budget saves via PATCH. No 405 error. Persists on reload	PASS

MT-416	Switch client in dropdown	Settings reload for new client	Form clears, new client budget populates without stale data flash	PASS

Exports (FR-06)

Test ID	Action	Expected	Actual	Status
MT-417	Export CSV with Anderson selected	CSV contains Anderson-only data	CSV downloads with 20 columns but contains unfiltered full dataset	FAIL — DEF-019
MT-418	Export PDF with AWS filter active	PDF reflects active filters	PDF downloads but contains unfiltered data	FAIL — DEF-018
MT-419	Export CSV after DEF-018/019 fix	CSV respects active filters	CSV contains only filtered data matching active selection	PASS (retest)
MT-420	Export PDF after fix	PDF respects active filters	PDF contains only filtered data	PASS (retest)
MT-421	Spot-check CSV column count	20 columns present	All 20 columns: Date, Client, Provider, Service, Type, Unit, Units Used, Unit Cost, Total Cost, Daily Avg, 30d Trend, Forecast, AWS/Azure/GCP Rate, Cheapest Provider, Switch Savings, Util %, Status, Est Monthly Savings	PASS

Defects Found: DEF-018 (PDF export unfiltered), DEF-019 (CSV export unfiltered), DEF-020 (Recommendations missing client), DEF-021 (Waste shows Multiple instead of client name)

Session 5: Final Verification & Regression (Feb 15)

Purpose: Verify DEF-020/021 fixes and complete final regression before PR #31 submission.

Test ID	Action	Expected	Actual	Status
MT-501	Recommendations with client selected	Client name on each card	Anderson Consulting attribution displays on all recommendation cards	PASS
MT-502	Waste alerts with All Clients selected	Individual client names per alert row	Client names resolve correctly per alert row (not Multiple)	PASS
MT-503	Full dashboard load (All Clients, Last 30 Days)	All cost cards, trend chart render	Complete dashboard loads with AWS + Azure + GCP totals. ~2–3s load time (DEF-015)	PASS
MT-504	Budget UI hidden for All Clients	Budget elements not shown in aggregate	Budget bar and threshold indicators hidden when no specific client selected	PASS
MT-505	E2E: Anderson → Waste → Recs → Export CSV	Complete user workflow	Full flow completes: select client, review waste, check recs, export filtered CSV	PASS

Defects Discovered During Manual Testing

ID	Session	Description	Root Cause	Resolution	Status
DEF-008	1	Client dropdown empty	Response parsing mismatch	Fixed key extraction + formatter	Closed
DEF-009	1	GCP costs not displayed	Missing provider card	Added GCP card + styling	Closed
DEF-010	1	Trend chart single data point	Fetch limit=100 of 40K	Increased limit to 50,000	Closed
DEF-011	1	Cards stacked vertically	CSS grid not applied	4-column layout fix	Closed
DEF-012	1	FR-08 no UI section	DOM element never built	Built metrics UI section	Closed
DEF-013	1	Provider filter no data	IDs: 1,2 vs 2001,2002,2003	Corrected PROVIDER_IDS	Closed
DEF-014	2	Provider IDs + client name	Config IDs wrong + LastFirst	Updated IDs + formatter	Closed
DEF-015	4	Initial load ~2–3s	50K rows on page load	Enhancement (open)	Open
DEF-016	2	Service filter shows inactive	Dropdown from services table	Filtered to active services	Closed
DEF-017	3	Budget save 405 error	POST instead of PATCH	Switched to PATCH method	Closed
DEF-018	4	PDF export unfiltered	Used full dataset	Synced with active filters	Closed
DEF-019	4	CSV export unfiltered	Used full dataset	Synced with active filters	Closed
DEF-020	4	Recs missing client name	No client context passed	Added client attribution	Closed
DEF-021	4	Waste shows Multiple	No client grouping	Added client name resolution	Closed

February 17, 2026

Manual Test Coverage by Functional Requirement

FR	Requirement	Manual Tests	Pass	Fail→Fix→Repass	Final Status
FR-01	View Cost Dashboard	Steps 1,15; MT-401,503	4	0	Complete
FR-02	View Spending Trends	Steps 4-5	0	2→DEF-010→Pass	Complete
FR-03	Filter by Provider/Service/Date	MT-201–207, MT-301–302	8	2→DEF-013,016→Pass	Complete
FR-04	View Waste Alerts	MT-403–408, MT-502	7	1→DEF-021→Pass	Complete
FR-05	View Recommendations	MT-409–413, MT-501	5	1→DEF-020→Pass	Complete
FR-06	Export Reports (CSV/PDF)	MT-417–421	3	2→DEF-018,019→Pass	Complete
FR-07	Set Budget Thresholds	MT-414–416	3	1→DEF-017→Pass	Complete
FR-08	View Resource Metrics	Steps 2,16	0	2→DEF-012→Pass	Complete

All 8 functional requirements verified via manual testing. 14 defects found, 13 resolved, 1 open (enhancement).