MADELABS
TECHNOLOGY

# Vessel Vanguard

## VV360
## Architecture Remediation

Statement of Work

12/03/2025

# Contents

# Introduction

MadeLabs is pleased to submit this Statement of Work to Vessel Vanguard in response to a request from Brian Hagan and Dan Roberts. This Statement of Work describes the activities to be performed to remediate architectural, performance, and data quality issues of VV 360 and is based off of the findings during the Architecture and Performance Assessment conducted.

This is an estimate, and though we do our due diligence in estimating software development it is notoriously fluid based on changes in scope as new opportunities or issues present themselves.

## Executive Summary

This Statement of Work addresses critical performance, data integrity, and architectural issues identified in the MadeLabs technology assessment of the Vessel Vanguard 360 platform. The current architecture, while functional, faces scalability challenges that will compound as the customer base and data volume grow.

### Key Problems to Solve

1. **Slow Application Performance** - Users experience long load times due to the "offline-first" architecture loading all data upfront
2. **Data Loss Risk** - Multiple scenarios exist where user data can be silently lost and become out of sync.
3. **Database Mismatch** - DynamoDB is not suited for the application's relational data patterns
4. **Admin Inefficiency** - Administrative functions are severely impacted by performance issues
5. **Scalability** - The current application will be challenged to scale over time with more customers, vessels, and records being added.

### Team & Capacity

- **Team Size:** Equivalent of 2 full-time developers
- **Weekly Capacity:** 80 hours
- **Average Monthly Capacity:** ~347 hours (80 hrs × 4.33 weeks)
- **Skillsets Included:** Architecture, Development, AWS Cloud Infrastructure, UI/UX, Testing, Project Management

### Billing Structure

- **Rate:** $115/hour
- **Monthly Investment:** $40,000 (fixed monthly)
- **Billing Cycle:** Consistent monthly amount

### Working Model

This remediation capacity is dedicated and additive to existing team resources. The 80 hours/week is allocated specifically to this SOW while separate resources continue to support:

- Ongoing production application support
- Continued feature development
- Bug fixes and maintenance

The remediation team will work collaboratively with existing team members to ensure alignment and smooth integration of changes.

### Key Assumptions

- Single-device offline functionality is the target (onboard server for multi-crew collaboration is out of scope)
- 80 hours/week capacity is fully dedicated to this remediation work
- Timeline estimates assume consistent availability and no major scope changes

## Activities and Deliverables

During this focused engagement, MadeLabs will work closely with Vessel Vanguard to ensure that all work and deliverables are provided in a timely manner in accordance with this agreement as outlined below.

MadeLabs will provide architecture, engineering, dev-ops, project management and UI/UX design services as part of this engagement.

## Scope of Work

### Phase 1: Onboarding & Quick Wins

**Objective:** Get new developers set up and ready to contribute with a focus on reducing immediate user pain points and reduce data loss.

1.1 Data Loss Reduction

Implement safeguards to prevent users from losing unsynced data:

- **Sync status indicator** - Visual component showing pending changes and sync state
- **Logout guard** - Prevent logout when unsynced changes exist; warn user with override option
- **Vessel switch guard** - Block vessel switching until changes are synced or user acknowledges

- **Admin organization guard** - Same protection for admin users switching between organizations

1.2 Error Handling

Add comprehensive error handling across all data operations:

- **DataStore operation wrapper** - Standardized try/catch handling for all save/delete operations
- **User notification system** - Surface errors via toast notifications using existing middleware
- **Error logging** - Capture errors for debugging and monitoring

1.3 Admin Performance

Improve admin responsiveness by changing data loading patterns and interaction with the API:

- **Pagination of lists** - Load vessels in pages rather than all at once.
- **Lazy-loaded detail views** - Fetch related data (tasks, equipment, etc.) only when needed
- **Direct GraphQL queries and mutations** - Bypass DataStore for admin operations

Admin functionality discovered includes: vessels, users, organizations, templates, enrollments, and a variety of tools that we'll need to decide together on whether are in use or should be excluded; for example: business coupons, invoices, leads, payments, and products.

## Phase 2: Offline Strategy Redesign

**Objective:** Right-size offline capabilities to match actual business needs

2.1 Offline Requirements Definition

*Requires VV stakeholder workshop to resolve:*

- For web users: What data is critical for offline use?
- For mobile: What data is critical when at sea?
- Current vessel only vs. multiple vessels?
- Historical data retention period?
- Which features must work offline?

2.2 Selective Sync (Web & Mobile Users)

Implement user-controlled offline data selection to reduce initial load times:

- **Vessel selection interface** - Allow users to choose which vessel(s) to sync for offline use
- **"Prepare for offline" workflow** - Guided process to download selected data before going offline

- **Offline status dashboard** - Show what data is available offline and last sync time
- **Storage management** - Clear offline data and manage storage
- **Optimized sync expressions** - Refine DataStore sync to only pull selected vessel data (not all vessels)

2.3 Admin Application - Online-Only

Remove DataStore dependency for admin functionality (admins always work online):

- **Direct GraphQL queries** - Admin operations use API calls instead of DataStore
- **Server-side pagination** - Load data in pages with server-side filtering
- **No local persistence** - Admin data not stored in IndexedDB
- **Separate data layer** - Admin components use different data fetching pattern than user components
- **AWS WAF protection** - Configure Web Application Firewall rules to restrict admin access (IP allowlisting, rate limiting, bot protection)

## Phase 3: Database Migration (DynamoDB to PostgreSQL)

**Objective:** Establish a relational database foundation that matches application data patterns

3.1 Database Design & Setup

Design and provision the new database infrastructure:

- **PostgreSQL schema** - Design relational schema based on existing entity relationships (documented in assessment)
- **Aurora cluster provisioning** - Configure AWS Aurora PostgreSQL for dev, staging, and production environments
- **Security configuration** - Connection pooling, IAM roles, VPC security groups, encryption
- **Backup procedures** - Point-in-time recovery, automated backups, disaster recovery runbook

3.2 Data Access Layer

Build optimized data access patterns:

- **Repository layer** - New data access implementation for PostgreSQL with connection management
- **Query optimization** - Resolve N+1 patterns with proper JOINs and batch loading
- **Indexing strategy** - Create indexes for common query patterns; document index decisions
- **Performance benchmarks** - Measure before/after query performance

3.3 GraphQL Resolver Migration

Update API layer to use new database:

- **Resolver rewrites** - Migrate all GraphQL resolvers from DynamoDB to PostgreSQL
- **DataLoader implementation** - Batch loading to prevent N+1 at the API layer
- **Lambda updates** - Update backend Lambda functions for new data layer
- **API compatibility testing** - Verify existing clients work without changes

### 3.4 Data Migration

Execute phased migration of production data:

- **Migration scripts** - Per-entity scripts with validation and error handling
- **Validation tooling** - Compare source and target data, report discrepancies
- **Migration runbook** - Step-by-step execution plan with rollback procedures
- **Parallel operation** - Maintain DynamoDB as fallback during transition period

**Proposed Migration Order (by risk/complexity - subject to change):**

1. Templates & Release Notes (lowest risk)
2. Testing & Sensors
3. Media & Documents
4. Communication & Notifications
5. Billing & Sign Up
6. Safety Management System (SMS)
7. Maintenance & Work Management (highest complexity)

## Phase 4: Conflict Resolution System

**Objective:** Provide transparency and user control when data conflicts occur

### 4.1 Conflict Detection

Implement server-side conflict identification:

- **Detection Lambda** - Server-side function to identify conflicting changes during sync
- **Version tracking** - Track record versions and modification timestamps
- **Conflict queue** - Store conflicts for user resolution rather than auto-resolving

### 4.2 Conflict Resolution UI

Give users visibility and control over conflicts:

- **Resolution interface** - Side-by-side comparison of conflicting versions
- **Merge workflow** - Allow user to choose winner, merge fields, or discard
- **Audit logging** - Record all conflict resolutions for compliance or future debugging

- **User documentation** - Help content explaining conflicts and resolution options

### Phase 5: Historical Data Management

**Objective:** Implement data retention policies to maintain performance at scale

5.1 Archival Strategy

Establish policies and automation for data lifecycle:

- **Retention policies** - Define retention periods by entity type (with VV input for compliance needs)
- **Archival automation** - Background jobs to move aged data to archive storage
- **Archive access** - Interface for retrieving historical data on request (within 24 hours SLA)
- **Compliance documentation** - Document retention policies for audit purposes

# Roles and Responsibilities

## Vessel Vanguard Responsibilities

Vessel Vanguard shall afford MadeLabs access to the necessary staff at various times throughout this engagement to allow MadeLabs to fulfill the commitment of the services specified. MadeLabs will make every reasonable effort to limit the demands on Vessel Vanguard personnel's time.

Obligations:

- Assign a primary point of contact from Vessel Vanguard to manage the engagement (currently, this is Brian Hagan).
- Notify MadeLabs of any changes to key personnel from Vessel Vanguard who are responsible for the engagement.
- Provide appropriate access to systems, networks, tools, or personnel to fulfill the commitment of services specified.
- Ensure timely review and feedback of all deliverables provided by MadeLabs to fulfill the commitment of services specified.

## Mutual Responsibilities

In support of the success of this engagement:

- Both parties will conduct meetings at mutually agreed upon times.
- Support and provide sufficient representation at these meetings to move the work and deliverables forward.
- Collaborate to adjust the schedule in the event of a newly discovered requirement, change, or significant technical discovery.

- Meet periodically throughout this engagement, either in person or through electronic means, to ensure expectations are being met, resolve any issues, and plan for the next phase of development.

### MadeLabs Responsibilities

- Provide qualified resources to complete the upgrades and development and provide recommendations on the current state of development and future iterations.
- Perform the duties as outlined in the Activities and Deliverables.
- Provide, at minimum, weekly progress updates in addition to ad-hoc discussions regarding work.

## Estimated Schedule

| Phase | Duration | Est. Hours | Investment |
|---|---|---|---|
| **Phase 1:** Onboarding & Quick Wins | 3-4 weeks | 240-320 hrs | $27,600 - $36,800 |
| **Phase 2:** Offline Strategy Redesign | 4-5 weeks | 320-400 hrs | $36,800 - $46,000 |
| **Phase 3:** Database Migration | 6-8 weeks | 480-640 hrs | $55,200 - $73,600 |
| **Phase 4:** Conflict Resolution | 3-4 weeks | 240-320 hrs | $27,600 - $36,800 |
| **Phase 5:** Historical Data Management | 2-3 weeks | 120-160 hrs | $13,800 - $18,400 |
| Testing, Refinement and Support | Remaining time and approved budget | | |

# Location

MadeLabs will conduct all work remotely.

# Fees and Expenses

This engagement is based on **time and material for an estimated 4-6 months**. MadeLabs resources will be used to leverage various skill sets based on the agreed-upon scope at a discounted rate of $115/hour. The total estimate ranges between $160,000 to $240,000.

Any change in scope will be discussed beforehand and agreed upon by all parties. *Scope can be reduced to directly reduce cost*. Any changes that will increase the scope or timeline will be approved by Vessel Vanguard prior to conducting work if not already specified in this document.

Vessel Vanguard will be invoiced each month for work conducted during that month and shall pay within 30 days of invoice.

## Summary

| Metric | Estimate Range |
|---|---|
| **Total Duration** | 4-6 months |
| **Total Investment** | $160,000 - $240,000 |
| **Monthly Run Rate** | $40,000/month |

# Appendix

## A. Reference Documents

- MadeLabs Assessment Report (November 12, 2025)
- VV 360 Current Architecture Diagrams (see assessment)
- Entity Relationship Diagrams (see assessment pages 8-14)

## B. Glossary

- **DataStore** - AWS Amplify's offline-first data synchronization library
- **N+1 Query** - Anti-pattern where fetching N records results in N+1 database queries
- **Aurora PostgreSQL** - AWS managed PostgreSQL-compatible relational database
- **Selective Sync** - Synchronizing only user-selected data subsets for offline use
- **Conflict Resolution** - Process of resolving differences when the same data is modified in multiple places

# Signatures

## Agreement

MadeLabs and Vessel Vanguard agree to the Statement of Work outlined above and the associated costs.

Accepted By:

| Madelabs LLC | Vessel Vanguard |
| --- | --- |
| By: _____ | By: _____ |
| Print Name: _____ | Print Name: _____ |
| Title: _____ | Title: _____ |
| Date: _____ | Date: _____ |