CASE STUDY: New High-Performance Data Acquisition System

From Decades-Old Bottleneck to Dynamic, Test-Engineer-Driven Tool in 12 Weeks

Project Snapshot

Industry: Aerospace / Combustion Testing
Challenge: Legacy data acquisition system created delays, data-loss risks, and dependency on one specialist, costing tens of thousands weekly in lost productivity.
Result: A modern, customizable GUI platform with auto-saving, Excel integration, and on-the-fly configuration—delivered as a minimum viable product in 3 weeks, fully functional in 12 weeks.

---

1. Opening Hook – The Challenge

A major test facility relied on a **decades-old data acquisition system** managed by a single specialist. While it collected basic data, it couldn't keep pace with evolving test requirements or late-shift operations.

Engineers faced:

- **No ability to make changes** without the specialist on-site.
- **Static executables** requiring recompilation and redeployment for any tweak.
- **Critical calculations** becoming inaccurate mid-test, with no quick fix.
- **Data-loss incidents** from forgotten manual "record" clicks.

These constraints translated to **tens of thousands of dollars lost weekly** in stalled testing, delayed insights, and wasted cell time.

---

2. Problem Definition

We interviewed multiple test engineers to document pain points:

1. **Inflexible GUI**: No way to add indicators, images, or panels dynamically.
2. **Static calculations**: Advanced formulas couldn't be edited without recompiling the executable.

4. **Single-point dependency**: One specialist bottlenecked all updates.
5. **Manual data saving**: Risk of losing entire test runs due to human error.
6. **Limited data sourcing**: Couldn't easily integrate new sensors or network sources.
7. **Slow iteration**: New test ideas couldn't be implemented without days of delay.

The goal was a **test-engineer-driven system** that was fast to configure, always saving data, and extensible to advanced calculations—all without specialized programming skills.

---

3. Our Approach

Step 1: Requirements and Minimum Viable Product (MVP)

We partnered with a controls expert to build an **MVP in just three weeks**—a functional data logger with core GUI customization. This demonstrated feasibility and gathered immediate feedback from users.

Step 2: Rapid Iteration (7 Additional Weeks)

We systematically addressed each pain point:

Dynamic GUI Customization

- **Drag-and-drop indicators**: Add tens of channels in minutes.
- **Custom panels**: Import images, adjust layouts, align groups, format numbers—all on-the-fly.
- **No recompilation needed**: Changes take effect instantly.

Advanced Calculations with Excel Integration

- Connected the platform to **Excel**, enabling all existing advanced add-ins and formulas.
- Test engineers could **edit calculations live**—input a new formula, and it updates in real-time.
- This preserved their library of complex combustion analytics without rewriting code.

Always-On Data Management

- **Automatic continuous saving** eliminates manual "record" clicks.
- No more data loss from forgotten buttons—every run is captured from the moment it starts.

Expanded Data Sourcing

- Pulls from **networked devices** and **hardwired sensors** seamlessly.
- Debugging tools for troubleshooting new integrations.

Total Timeline: 12 Weeks

From initial interviews to a **production-ready, bulletproof system** that empowered test engineers to customize without external help.

---

4. Results and Impact

Capability	Legacy System	New Platform
GUI Customization	Static, recompilation required	Drag-and-drop, instant
Calculations	Baked-in, uneditable	Live Excel integration
Data Saving	Manual button	Automatic continuous
Update Speed	Days (specialist needed)	Minutes (self-service)
Data Sources	Limited	Network + hardwired
Engineer Autonomy	Single specialist	All test engineers

Business Impact:

- **Recovered thousands weekly** in test-cell productivity.
- **Eliminated data-loss risk**, saving costly re-runs.
- **Accelerated test iteration**—engineers now implement ideas during shifts.
- **Scalable platform** supports future sensors and analytics without bottlenecks.

---

5. Takeaway & Forward Value

This project exemplifies **ProReady Engineering's approach** to technical bottlenecks:

- **Deep user interviews** to document real pain points.
- **Rapid prototyping** (MVP in 3 weeks) to prove value fast.
- **Clever integrations** (Excel for calculations) to leverage existing assets.
- **Empowerment over dependency**—tools so intuitive that test engineers become self-sufficient.

delivered a system that's **dynamic, reliable, and future-proof**, turning a legacy straint into a competitive advantage for the test team.