

Agile Software

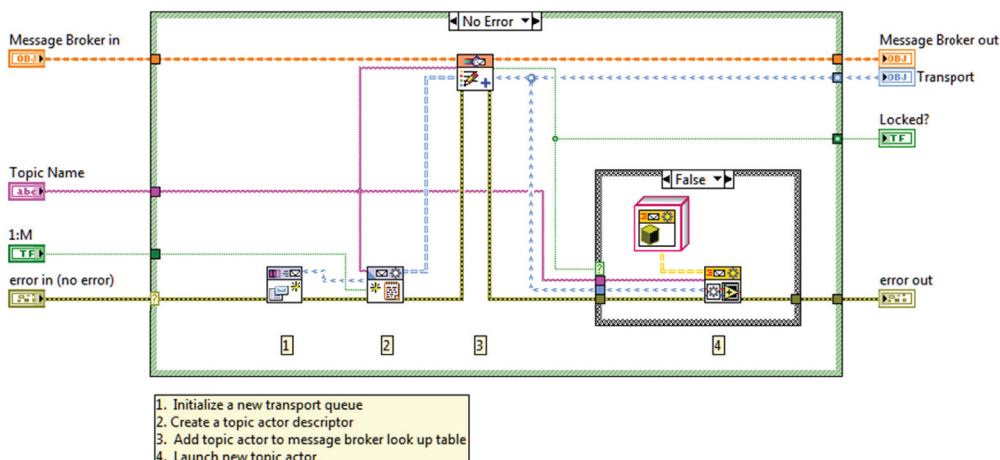
Sprint through your new product development with Bloomy!

Are you developing a new software product on an aggressive schedule? Do you need rapid progress and the flexibility to design your product's software in parallel with hardware? Are your requirements not fully defined? If so then an agile approach to project management coupled with SOLID¹ software design is right for you. At Bloomy we use LabVIEW in combination with Robert Martin's five SOLID principles of software design, and an agile scrum approach to project management. These techniques allow us to deliver the software you need, on time and on budget.



SOLID Development

SOLID software development makes your software easier to manage and less risky to maintain. We believe that software should be “soft,” not brittle. Try as we might, we never know all requirements at a project's start, but developing SOLID code lets us handle the inevitable changes in stride. SOLID software's main value is its ability to gracefully adapt to change. With SOLID software development Bloomy delivers maximum long-term value. If you care about controlling budget, advancing schedule, reducing risk, avoiding complex code rewrites, and simplifying regression testing then SOLID is for you.



Bloomy leverages LabVIEW object-oriented design to quickly architect modular code that readily adapts to our customers' changing requirements. The diagram to the left is a sub-function of a larger event aggregation pattern that implements a dynamic messaging framework. This framework allows zero coupled messaging between actors in an actor-oriented application.

Developing SOLID software with agile project management lets Bloomy deliver incredible software value faster and with reduced project risk. By following the five principles of SOLID software design we are able to write more modular software that will scale with the changing needs of the product owner. Let us show you how we use these industry-proven software development techniques to solve your most challenging problems.

Agile Project Management

Agile project management is an iterative method of managing the design, coding, and testing for new product development projects in a highly flexible and interactive manner. The agile process does not just handle change throughout a project's lifetime but actually embraces it. If you have an idea of where you need to be one year from now but don't have the luxury of defining every detail before starting the project then an agile approach to project management is an excellent way to get started.



At Bloomy we use the agile methodology known as scrum for software projects with highly emergent requirements. The scrum methodology allows Bloomy to deliver working software to the product owner in two week development cycles known as sprints. Between sprints the product owner has the ability to evaluate the working software and make decisions on where next to focus development efforts. By following the scrum process Bloomy is able to adapt our development efforts to meet the changing needs of the product owner, making scrum the process of choice for complex software projects.

Communication

An important aspect of agile development is maintaining a tight communication loop with you, the product owner. We accomplish this by including the product owner in all of our scrum ceremonies. This inclusion helps keep you informed about the team's progress and makes you aware of technical challenges the team faces in real time. Collaboration of product owner and development team is highly encouraged when feasible.

“ IDEX Health and Sciences teamed up with Bloomy to create a dynamic integrated development environment on top of National Instruments LabVIEW for our Fluidcis Development Kit product. Bloomy's use of Agile project management allowed us to reprioritize our user stories as we progressed through the sprint cycles, enabling us to successfully innovate as we developed. This flexibility maximized the amount of collaborative creativity the entire team was able to inject into the software, which is a large part of why it has been so successful. ”

John Fulton
Project Engineer IDEX Health and Sciences

1 SOLID stands for Single responsibility, Open-closed, Liskov substitution, Interface segregation and Dependency inversion, which are the five basic principles of object-oriented programming and design.

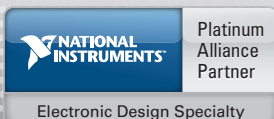
Single responsibility - a class should have only one reason to change

Open-closed - software entities should be open for extension, but closed for modification

Liskov substitution - objects in a program should be replaceable with instances of their subtypes without altering the correctness of that program

Interface segregation - many client-specific interfaces are better than one general-purpose interface

Dependency inversion - one should depend upon abstractions and not upon concretions



Call 860-298-9925 or visit
www.bloomy.com