

## Introduction

LBi utilizes an agile software development methodology that focuses on the user and incorporates components from various standard methodologies. We have the flexibility to tailor the specific methodology to the project at hand. This document describes the methodology we use as a starting point in a new project.

The LBi methodology combines elements from Extreme Programming (XP), Agile Unified Process (AUP) and Rational Unified Process (RUP). We leverage the customer and stakeholder focus of XP, artifacts of AUP and the attention to detail of RUP. In addition we emphasize an in-depth understanding of the business of the client/user throughout the project, from inception to transition.

## Our User-Centric Mindset

At the core of LBi's approach to all projects (whether support, documentation, training or development) is our user-centric mindset. This mindset has two major rules - (1) Know the Business and (2) Keep the user involved throughout the project.

For example, let's imagine we were given the assignment of developing a new Point of Sale (POS) application. What would you do first? At LBi, we'd learn the business. We would discover that the application would have to do things like suspend transactions (the customer has to go back and get "one more thing"), back out an item, and re-total when the customer points out that the clerk left an item out or wants to impulsively add another item. These functions, intrinsic to POS systems, may have significant impact on the overall system design.

Now the problem here is that most of us want to just get going with the design and programming and testing. Few have the desire to patiently sit with clients and listen to them somewhat haphazardly provide requirements. At the same time it would be arrogant of us (as well as naïve) to believe that a week or a month or even months of learning the business and developing high level requirements and functionality will provide us with everything we need to develop the system. So we have to build in to our process a mechanism that will uncover what elements of the design that we have ignored, missed or misunderstood. This mechanism is involving the user in each step of the process, from the project start of "learning the business" right through production. The user should be part of the team.

## LBi's Methodology

Step 1 Inception - Our first step is to identify application scope and learn the business environment. We must also understand the motivation for developing the application, its expected growth, compliance issues, and any Service Level Agreements.

- a. Learn the business – Sit with users, review existing documentation and learn the required tasks. This is the most important step as with this understanding of the client's challenges the new application will be designed to improve their process.
- b. Identify initial scope of project
- c. Set the project goals and objectives.

Step 2 Elaboration - Next we select appropriate architecture and prototype the interface. We also separate the project into small “bite-sized” functional units. Depending on the application/project this may be modules, sub modules, reports, and/or screens. Each unit must perform a function that can be delivered to the user

- a. Select appropriate architecture and application framework
- b. Prototype the interface
- c. Separate the project into small “bite-sized” function units. This decision may impact the architecture, so a. and b. may be repeated.

It is important to note that at no time during this early phase of the project or, for that matter, any phase of the project do we operate “in a vacuum”. The user is integral to the active process. The purpose of the prototype at this early stage is to engage the user and pique their interest. The reward is the reality check of the user’s feedback at an early stage, perhaps averting a catastrophe in the application framework. The application’s design must be extremely flexible in order to adapt to changes in requirements, thus minimizing the cost impact. Throughout the entire development cycle we understand there will be changes which will require system adaptation

Step 3 Construction - We now perform the following steps for each functional unit (small “bite-sized” delivery), where each delivery has its own mini-lifecycle. Once again, feedback from the user is encouraged at all times. These are iterative steps that are repeated over the lifecycle of the project.

- a. Requirements Gathering – user meetings, use cases
- b. Preliminary Design – workflow, sequence diagrams
- c. Prototyping
- d. Specification and Design – class diagrams and modeling
- e. Development and Testing
- f. Delivery to staging area
- g. QA

Step 4 Transition – This includes any final QA and delivery.

- a. QA process – regression testing
- b. Delivery to production environment

Depending on the project and client there may be variations to the above approach. For example, the client may require the application to be completed before it goes live or preferably move each functional unit into production as it is ready. The ultimate user feedback is their use of the application in the real world.

## The Benefits to the LBi Approach

The key benefits to our user-centric approach are:

- Engages users throughout the project lifecycle
- User becomes familiar with the system earlier in the lifecycle
- Shortens development cycles as users are able to see the product earlier and make desired changes before the application is delivered
- Encourages user feedback
- In the end the user gets an application that will maximize business value
- Our flexible design and attitude encourages earlier recognition of mistakes
- Reduces training
- Avoids large costly change requests at go live