FaultTrack ALM DCOM Engineering, LLC.

Software Construction Document  
This document outlines the plan for software construction that will be undertaken to develop  
the FaultTrack ALM project.

David Anderson  
Principal Software Development Engineer  
 Design. Develop. Share.

Software Description

FaultTrack ALM is a shortened name for FaultTrack: Application Lifecycle Management, and is planned to be an issue tracking and project management system for software development projects.

**Feature Overview**

FaultTrack ALM will be broken down into the following feature sets and areas.

* Project Management
* Iteration / Sprint Management for Agile development practices
* Team Management and Collaboration
* Build and Release Management
* Documentation Management

Software Concepts

**Project Collection**

A project collection is a named group that contains a list of projects.

* A project collection is the top most logical item in the project hierarchy; it is the root item
* A project collection must have a unique name
* A project collection may contain a list of zero or more projects

**Project**

A project is a representation of a real-world project that has the concept of a version number and generally has a team of people assigned to it.

* A project must belong to a project collection
* A project can only belong to one project collection
* A project may contain a list of zero or more other projects
* A project must have a unique name within the project collection or project that it belongs to
* A project may contain a list of zero or more project versions

**Project Version**

A project version is a specific release of a project that contains a list of issues and features.

* A project version must belong to a project
* A project version can only belong to one project
* A project version must be unique within the project that it belongs to
* A project version may contain a list of zero or more features
* A project version may contain a list of zero or more issues

**Feature**

A feature is a requested or approved modification or enhancement to a project.

* A feature can only to one project, but a copy of the feature can belong to another project
* A feature does not have to be unique because it can exist in many places
* A feature should have the following properties
  + Short Description
  + Long Description
  + Status

**Sprint / Iteration**

A sprint, or iteration, is a fixed length of time where a team commits to a certain amount of work to complete during that period. Common sprint and iteration lengths are two weeks or one month, but can vary depending on project size, team size, and the amount of work. The core of a sprint or iteration is part of the Agile development process, and focuses in deliverable software with a “definition of done” at the end of each iteration. In terms of FaultTrack, a iteration would consist of tickets, whether it be bugs or features or tasks.

The major problem that FaultTrack solves is that iterations can be multi-project and multi-version. It’s seen only as a time-boxed period of committed work. Whereas most project management or ticket systems restrict teams to iterations that only work for specific versions of specific projects. For multiple-project, or smaller teams, this is a large issue because the overhead of managing multiple iterations consumes quite a large amount of time, when the software could instead be made flexible enough to handle this well.

N-Tier Architecture

**Microsoft SQL Server**

A Microsoft SQL Server database will be the technology of choice for persistence.

**Microsoft Entity Framework ORM**

Microsoft Entity Framework will be used as the ORM of choice to abstract the persistence layer away. POCO’s will be used to represent the domain model entities.

**DDD (Domain Driven Development)**

A DDD oriented pattern will be used to abstract away the ORM layer from the outside world. Data Transfer Objects (DTO’s) will be used for transferring data between the service layer and the client layer boundaries.

**Microsoft Windows Presentation Foundation using MVVM**

The Model-View-ViewModel pattern will be used with WPF for the client. Model’s and ViewModel’s will be used to represent the data and behavior for the UI.

