Bethany Beery

CIT275 – Instructor Jeff Straw

Northwestern Michigan College

DESIGN PATTERNS

Design patterns are general reusable solutions to common reoccurring problems in code structure. They are not a literal implementation but rather an instructional guide to address an issue of object generation and interaction. Patterns can speed up the design process by providing proven solutions.

Christopher Alexander popularized the concept of patterns in the context of architecture in the late seventies. Just over ten years later in 1994, four programmers applied Alexander’s concepts of pattern reuse to software architecture releasing a book titled, Design Patterns: Elements of Reusable Object-Oriented Software. These four authors are collectively known as Gang of Four (GOF). The book became widely accepted in the industry, becoming a practice everywhere.

Design patterns are composed of twenty-three different patterns which can be divided into three categories; creational, structural, and behavioral. The focus of creational patterns is class instantiation. Patterns within the creational category use objects to create instantiation rather than instantiate directly. Structural patterns group objects together into larger structures to obtain new functionality. Inheritance is used to compose interfaces. The third category, behavioral patterns, is about the communication between objects.

Patterns also provide a point of reference amongst professionals. Rather than try and explain architectural decisions, one can simple name a pattern. It can take years of experience to learn when and how to most appropriately apply patterns to solve problems.