

Fach: Verteilte Informationsverarbeitung

Studienheft: VII01-03

Projekt: Client-Server-Malprogramm

Dokumenttitel: Phase 5: Wartung und Pflege

Dokumentart: -

Erstellt von: Martin Blankenstein

Erstelldatum: 26.08.2010

Inhaltsverzeichnis:

1 Rückblick auf Projektdurchführung.....	2
2 Zukunftsausblick.....	2

## 1 Rückblick auf Projektdurchführung

Die Durchführung des Projektes hat anfangs viel Spaß gemacht. Die Ansatz der Event-Based-Components ist interessant und die Verwendung kann man mit etwas Übung beschleunigen. Die von mir für dieses Projekte selbstgesteckte Bearbeitungszeit von 40 Zeitstunden habe ich mit einem realen Zeitaufwand von 75 Zeitstunden leider nicht eingehalten. Die Verwendung der EBCs kostete unerwartet viel Zeit, da besonders das Herunterschreiben der Nachrichtenklassen und Interfaces viel Zeit benötigt hat. Auch das Zusammenstecken der EBCs auf Platinen und die Verdrahtung ist zeitaufwändig, ebenso wie die Erstellung der Testfälle. Bei der Erstellung der eigentlichen Implementierung der vorher definierten Interfaces konnte man sich dann aber auf die Funktionalitäten beschränken, was bei der Entwicklung geholfen hat. Die Testfälle haben die Entwicklung ebenfalls positiv beeinflusst. Am Ende bleibt noch zu sagen das die Erstellung der Dokumentation rund ein drittel der Zeit beansprucht hat. Besonders die Entwurfsphase war sehr zeitintensiv.

Beim nächsten Mal würde ich einen etwas kleineren Umfang für meine Client-Server-Anwendung wählen und eine StartAnwendung wie den StartSelector nicht definieren, entwerfen, implementieren und testen. Das Paradigma der “Event-Based-Components” würde ich hingegen wieder verwenden. Bei öfterem Anwenden gelingt auch der Entwurf und die Implementierung der Nachrichten und Contracts schneller. Eventuell gibt es bald Programme mit denen man den EBC-Entwurf durchführt und die einem anschließend die Nachrichten, Contracts und Platinen automatisch generieren, so das man nur doch die Contracts implementieren muss.

## 2 Zukunftsausblick

Ein noch bestehendes und auf der TODO-Liste stehendes Problem ergibt sich der Verwendung der Start- und EndByteBlocks. Wenn der zu sendende Inhalt (z. Bsp. der aktuelle Malbereich) später durch Zufall einen dieser Blöcke enthält, dann gehen Nachrichten verloren. Deshalb müsste der Inhalt vor dem Senden die Start- und EndByteblöcke maskieren und der Empfänger wieder demaskieren. Da es sehr unwahrscheinlich ist, dass dieser Fehler überhaupt auftritt, bleibt die Maskierung ein noch zu behebendes, mögliches Problem.

Neben der Maskierung von Start- und EndByteBlock könnte die Maloberfläche noch um eine Toolbox mit verschiedenen Malwerkzeugen ausgestattet werden sowie der Möglichkeit die Farbe zu ändern.

Auch die Einbettung eines Chats, zum Beispiel unter dem Malbereich wäre möglich. Für diese Änderung müssten sogar nur wenige Änderungen vorgenommen werden.

Bei mehreren Clients die gleichzeitig malen, kann es unter Umständen sein das der Server überlastet wird. Ursache sind hier vor allem die vielen Lognachrichten die bei nur einem Malvorgang ausgeschrieben werden. Die Performance würde sich, durch das Entfernen der Lognachrichten die während des Malprozesses ausgeschrieben werden, verbessern.