

Wie der Name schon sagt sind wir spezialisiert auf das Finden von Lösungen. Denn durch unsere langjährige Erfahrung wissen wir, es ist nicht wichtig ob eine Lösung unter Windows oder Unix, C oder C++ entwickelt wird, wichtig ist, dass das Endprodukt das Problem löst und nicht nur verschiebt oder verschleiert. Und auch nicht immer muss die Lösung eines Problems Software sein, auch Umgestaltungen im Arbeitsablauf Ihres Unternehmens können sehr effektiv sein.

Darauf konzentriert sich loe.at und deckt dabei die gesamte Breite der IT Problemstellungen ab:

- **Analyse**

*Um welches Problem handelt es sich?
Wo ist der beste Angriffspunkt?*

Sehr oft werden Fehler bereits in dieser Phase des Projektes gemacht und können in späteren Phasen nur mehr unter hohem Einsatz von Ressourcen wieder gut gemacht werden. Deshalb ist es –vor allem in dieser Phase- wichtig mit der Realität in Kontakt zu bleiben, eine intensive „Beziehung“ zum Problem und seinem Umfeld zu bilden.

- **Konzeption**

*Wie sieht eine mögliche Lösung aus?
Was kann sie, was kann sie nicht?*

Hier wird eine mögliche Lösung grob beschrieben und alle Anforderungen und Einschränkungen dieser Lösung taxativ niedergeschrieben. Es ist wichtig, dass in dieser Phase noch keine Details der Implementierung oder der gewählten Umgebung vorausgesetzt werden, da dies erst nach dieser Phase entschieden wird da erst dann alle Anforderungen feststehen. Auch müssen diese beiden Phasen nicht vom selben Auftragnehmer erfüllt werden, womit der Auftragnehmer des Designs bereits in seinen Möglichkeiten künstlich eingeschränkt wird.

- **Entscheidungsunterstützung**

*Welche Lösung ist die beste?
Welche wird von den Mitarbeitern akzeptiert?*

Sehr oft steht das mittlere oder obere Management vor der Situation über ein IT- Projekt entscheiden zu müssen. Aber ein Teil der Mitarbeiter favorisiert Lösung A und ein anderer Lösung B. Hier kann oft die Meinung eines außenstehenden Dritten der entscheidende Input sein, um eine Entscheidung fällen zu können.

The solution is close...

- **Design**

*Welches Betriebssystem wird verwendet?
Welche Datenbank?*

In dieser Phase geht es darum, die Details der Implementierung festzulegen. Die Wahl der Umgebung und der verwendeten Technologien, sowie die Aufteilung auf Module erfolgt hier. Dieser Schritt wird durch die Verwendung von CASE Tools und genormten graphischen Darstellungsformen (UML) stark unterstützt.

- **Projektplanung**

*Welche Meilensteine gibt es?
Wie sieht der kritische Pfad aus?*

Ein Schritt der Hand in Hand mit dem Design geht, aber einen völlig anderen Blickwinkel einnimmt. Hier geht es vor allem darum, für das Projektmanagement Meilensteine als Kontrollpunkte zu definieren, sowie die Auslastung der Ressourcen zu planen.

- **Projektmanagement**

*Wird die Lösung wie geplant fertig?
Was sind zur Zeit die kritischen Punkte?*

Das Projektmanagement setzt sich aus zwei Teilbereichen zusammen: Dem Managen und dem Kontrollieren. Der Projektmanager muss dafür sorgen, dass alle Projektmitarbeiter die nötigen Informationen erhalten und auch dafür sorgen, dass Reibungen zwischen Mitarbeitern (oder auch Zulieferfirmen) möglichst früh erkannt und beseitigt werden.

Als Kontroller behält er die Übersicht über den Fortschritt des Projektes und auch über die bisher eingesetzten Ressourcen. Er hat die Verantwortung für die termingerechte Fertigstellung des Projektes. Sollten sich Verzögerungen oder ein erhöhter Ressourcenbedarf abzeichnen, so muss er mit dem Projektverantwortlichen gemeinsam eine Entscheidung treffen.

- **Implementierung**

*Habe ich genügend Ressourcen in meinem Unternehmen?
Habe ich bei meinen Mitarbeitern eine Kostengarantie?*

Natürlich bieten wir auch die Implementierung von Software an. Wir sind spezialisiert auf Serversysteme, egal ob unter Windows oder UNIX. Dabei verwende ich C/ C++/ Java, und zur Erstellung von kleinen Hilfsprogrammen bzw. Testprogrammen auch diverse Scriptsprachen.

Auch die Einbindung von Datenbanken oder anderen Varianten zur Datenspeicherung bilden kein Problem.

The solution is close...

- **Coaching**

*Was könnten meine Mitarbeiter besser machen?
Ist der Sourcecode wiederverwendbar?*

Programmieren kann man nicht lernen. Der einzige Weg besser zu programmieren ist zu programmieren! Und dabei kann man Mitarbeiter durch gezieltes coaching unterstützen. Denn vier Augen sehen mehr als zwei, und da es in der Programmierung nicht nur eine korrekte Lösung für eine Aufgabe gibt kann man hier weitere Möglichkeiten aufzeigen. Zusätzlich hat das Management die Meinung eines Aussenstehenden zu dem Code, der in Ihrer Firma produziert wird.

- **Qualitätssicherung**

*Funktioniert dieses Modul in allen Situationen?
Warum stürzt diese Applikation ab?*

Durch jahrelangen Umgang mit fremdem Sourcecode sind wir gewohnt Fehler in großen Systemen zu suchen ohne genau zu wissen in welchem Teil der Applikation er nun wirklich zu finden ist. Durch diese Erfahrung ist es uns auch möglich Testszenarien für Module und Applikationen zu erstellen, bzw. eine Analyse von Modulen auf ihre Fehlersicherheit hin durchzuführen.

- **Tuning**

*Warum ist die Applikation so langsam?
Kann die Applikation noch schneller beschleunigt werden?*

Die heutigen Programmiersprachen und Umgebungen machen es sehr leicht Applikationen zu entwickeln und dabei großen Fortschritt in wenig Zeit zu machen. Dabei wird aber oft auf die Performance vergessen. Da wir vor allem Echtzeitsysteme entwickelt haben sind wir auf einige Fallen sensibilisiert und können so ein bereits codiertes und getestetes Modul oft noch beschleunigen.

The solution is close...

Da ich sicher bin dass wir -nach dieser Fülle an Möglichkeiten einige Themen noch im Detail diskutieren werden, freue ich mich jetzt schon auf ein Meeting mit Ihnen.

Sie erreichen mich unter

loe.at IT & Logistics Solution-Finding
Mario Löffler
Eisteichgasse 25
8010 Graz
Österreich

Telefon: +43 664 2101210
Fax: +43 316 914151

www.loe.at
e-Mail: Mario.Loefler@loe.at