Projektbericht

Software Engineering Projekt

I8 | Erstellung und Verwaltung von Gruppen für Software Engineering Projekte

Namen

Datum

Inhaltsverzeichnis

1. Aufgabenstellung und Auftraggeber
2. Projektplanung
   1. Ausgangssituation
   2. Teamaufstellung und Rollenverteilung
   3. Kommunikation und eingesetzte Tools
   4. Dokumentation eingesetzter Tools
   5. Praktiken und Bewertungen
3. Projektdurchführung
   1. Iteration 09
   2. Iteration 10
   3. Iteration 11
   4. Iteration 12
   5. Iteration 13
   6. Iteration 14
   7. Iteration 15
   8. Iteration 16
   9. Iteration 17
4. Aufgabenstellung und Auftraggeber

Unser Team erhielt im Rahmen des Moduls Softwareengineering den Auftrag, eine bestehende Software zur Erstellung und Verwaltung von Gruppen für zukünftige Software Engineering Projekte weiterzuentwickeln. Ziel war es die Software zu optimieren und um neue Funktionen zu erweitern, um so eine noch effektivere Unterstützung bei der Bildung von Teams zu bieten. Dabei lag der Fokus auf der Weiterentwicklung der Algorithmen zur Bildung der Teams, sowie auf dem Gestalten einer benutzerfreundlichen Oberfläche. Unser Auftraggeber ist Herr Zirkelnach, der uns ebenfalls als Coach zur Seite gestellt wurde.

In der zweiten Phase der Softwareentwicklung ist der Gruppe bewusst geworden, dass der vorhandene Code erhebliche Probleme für die weitere Bearbeitung aufweist. Der Zeitaufwand, um den Code zu verstehen und zu bearbeiten, ist sehr hoch. Das bestehende Softwaresystem ist nur schwer erweiterbar aufgrund der Unübersichtlichkeit und der teilweise hart kodierten Struktur. Infolgedessen traf die Gruppe die Entscheidung, dass eine Neuentwicklung als die effizientere Lösung angesehen wird.

1. Projektplanung
   1. Ausgangssituation

Das Team bestand im ersten Modul von Softwareengineering aus 7 Mitgliedern. Die Zusammensetzung war wie folgt: fünf Wirtschaftsinformatikstudenten ( ), ein Wirtschaftsingenieurstudent ()und eine Informatikstudentin (). Planmäßig verabschiedete sich @KILLIANSCHINERLING nach SE l. Außerplanmäßig verließ @HIEU die Gruppe. Verstärkend wurde uns der Wirtschaftsinformatikstudent @VINCENTWEISE zur Seite gestellt. Glücklicherweise hatte dies an unserer Rollenverteilung nichts geändert. Die ausgeschiedenen Studenten hatten die Rolle der Analysten übernommen. Diese Rolle wurde von Phil Beck nun zusätzlich übernommen. Vincent Weise wurde verstärkend in die Rolle des Implementierers aufgenommen.

* 1. Teamaufstellung und Rollenverteilung

In SE l wurden die Rollen je nach Erfahrungen und Vorlieben verteilt. Dadurch, dass sich jeder in seiner Rolle wohlgefühlt hat, haben wir diese Aufteilung auch in SE ll übernommen und dem neuen Mitglied eine ebenfalls geeignete Position zugeteilt.

|  |  |  |
| --- | --- | --- |
| Name | Rolle | Verwendete Namenszeichen |
| Evelyn Dietrich | Projektmanager/ Deployment | ED |
| Nuha Alhamidi | Implementierer | NA |
| Julius Wyrembek | Tester | JW |
| Viktoria Karklin | Implementierer | VK |
| Phil Benedikt Beck | Analyse/ Tester | PB |
| Vincent Weise | Implementierer | VW |

* 1. Kommunikation und eingesetzte Tools

Zu Beginn von SE II gestaltete sich die interne Organisation in unserer Gruppe als recht unvorteilhaft. Wir trafen uns aller zwei Wochen zu einem Meeting, bei dem der genaue Termin für die nächste Iteration festgelegt wurde. Diese Vorgehensweise führte allerdings dazu, dass die Dauer der Iteration um plus/minus 3 Tage schwankte.

Obwohl diese Art der Terminfindung in SE l noch akzeptabel war, erkannte unsere Gruppe in der ersten Iteration von SE II, dass ein besseres System entstehen muss. Dies war dringend von Nöten, da wir uns erst am Anfang von SE ll darauf verständigt haben eine Neuimplementierung statt einer Weiterentwicklung durchzuführen. Dies hatte zur Folge, dass wir innerhalb derselben Zeitspanne mit einer größeren Anzahl an Aufgaben konfrontiert wurden. Dementsprechend beschlossen wir, unsere Iterationsdauer auf eine Woche zu begrenzen und uns alle zusammen wöchentlich am Mittwoch zur Aufgabenbearbeitung und Teambesprechung an der Hochschule zu treffen.   
Die Flexibilität war hierbei ein wichtiger Faktor. Es war nicht zwingend erforderlich, dass alle Teammitglieder permanent anwesend waren und sich gegenseitig bei der Bearbeitung der Aufgaben kontrollierten. Stattdessen praktizierten wir ein agiles Arbeitsmodell, bei dem jeder die Freiheit hatte, zu kommen und zu gehen, wann immer es ihm am besten passte. Allerdings war es von entscheidender Bedeutung, dass jeder seine zugewiesenen Aufgaben fristgerecht erledigte. Ein weiterer wichtiger Punkt war, dass alle Teammitglieder an den Teambesprechungen teilnahmen. Hier wurden Probleme und Herausforderungen diskutiert und Lösungen gefunden. Die Besprechungen boten auch die Möglichkeit, die Fortschritte der Gruppe zu besprechen und aufeinander abzustimmen. So konnten wir sicherstellen, dass alle Mitglieder auf dem gleichen Stand waren und dass wir als Team effektiv und zielgerichtet arbeiten konnten.

Generell konnten wir durch die neue Herangehensweise Fragen und Probleme direkt und schnell klären, was uns ermöglichte, effizienter zu arbeiten und schneller Fortschritte zu erzielen. Zudem waren alle Mitglieder der Gruppe gewissermaßen dazu gezwungen, kontinuierlich zu arbeiten, was uns als Team näher ans Ziel brachte.

Ein weiterer Vorteil dieser Methode war, dass jeder in der Gruppe den aktuellen Fortschritt der anderen direkt einsehen konnte. Dadurch konnten wir uns gegenseitig unterstützen und motivieren, was zu einer höheren Produktivität und besseren Arbeitsergebnissen führte.

Doch das Beste war, dass wir als Gruppe durch die Erfolge, die wir zusammen erreicht haben, zusammengewachsen sind. Wir konnten gemeinsam unsere Fortschritte und Erfolge feiern, was zu einem besseren Zusammenhalt und einer positiven Arbeitsatmosphäre beigetragen hat.

* 1. Dokumentation eingesetzter Tools
  2. Praktiken und Bewertung

Bewertung 1 sehr schlecht, 2 schlecht, 3 mittel, 4 gut, 5 sehr gut

|  |  |  |
| --- | --- | --- |
| Praktiken | Beschreibung | Bewertung |
| Weeklys | Eine Methode, bei der sich die Gruppenmitglieder einmal die Woche treffen um erzielte Ergebnisse und die nächsten Schritte zu besprechen. Damit wird sichergestellt, dass alle Mitglieder auf demselben Projektstand sind. | 5 |
| Risikobesprechung | Eine Methode zur frühzeitigen Reduzierung von möglichen Risiken. Dabei werden auch bereits vorhandene Risiken angesprochen und zeitnah eliminiert. (Risiken und Lösungsansätze sind in der Rikslist zu finden) | 4 – Erleichtere die Arbeit stark, jedoch ergeben sich wieder neue Probleme, die vorher nicht berücksichtigt wurden |
| Agile Entwicklung | Eine iterative und inkrementelle Entwicklungsmethode, die es Teams ermöglicht, schnell auf Änderungen zu reagieren und qualitativ hochwertige Software zu liefern. | 5 |
| Clean Code | Ein Ansatz, bei dem besonderer Wert auf lesbaren, wartbaren und gut strukturierten Code gelegt wird, um die Wartbarkeit, Qualität und Erweiterbarkeit der Software zu verbessern. Diese Entscheidung basierte auf dem Vorgängercode, da dieser ,unserer Meinung nach, die oben beschriebenen Eigenschaften kaum aufwies. | 5 |
| Lean Development | Eine Methode, die sich auf die Reduzierung von Verschwendung konzentriert, indem unnötige Schritte, Verzögerungen und Ressourcenverschwendung minimiert werden, um Effizienz und Produktivität zu steigern. | 4 - Einschätzung, ob Verschwendung vorliegt oder nicht, nicht immer gegeben |
| Pair Programming | Eine Methode, bei der zwei Entwickler zusammenarbeiten und den Code gemeinsam entwickeln, um die Qualität zu verbessern und das Wissen auszutauschen. | 4- Nimmt einen Ressourcen, da zwei Mitglieder an einer Aufgabe arbeiten |