

Samenvatting

De Unified Modeling Language (UML) is wereldwijd de de facto taal voor het modelleren van software systemen. De UML wordt gebruikt in alle domeinen van ontwikkeling van software systemen. De brede acceptatie van de UML is gedeeltelijk het gevolg van het feit dat deze taal flexibiliteit en vrijheid biedt bij het modelleren van software systemen: 1) UML biedt een uitgebreide verzameling van notaties voor het modelleren van een grote verzameling van concepten; 2) UML ondersteunt zowel meer formele als meer informele stijlen van modelleren.

In de context van model-driven software development, is een belangrijke open vraag hoe de kwaliteit van modellen beoordeeld moet worden. De mate van striktheid van modelleren is een factor die bepalend is voor de kwaliteit van een model. En het wordt aannemelijk geacht dat de kwaliteit van een model van invloed is op de kwaliteit van de uiteindelijke programma code.

Alhoewel UML steeds belangrijker wordt voor software ontwikkeling, mede door de opkomst van model-driven software development, is er vrijwel geen onderzoek naar de effecten van het gebruiken van UML modellen op software projecten. Deze vraag is met name interessant gezien de wijze op welke UML in de praktijk gebruikt wordt. Een van de belangrijke open vragen is of de wijze waarop UML gebruikt wordt van invloed is op de kwaliteit van de uiteindelijke programma code. Dit proefschrift beschrijft een serie empirische studies die gericht zijn op het beantwoorden van deze vraag. Bij deze studies worden verschillende onderzoeksmethoden gehanteerd (o.a. survey, experiment, en case studie) om deze vraag van verschillende invalshoeken te benaderen.

Een survey is uitgevoerd om te achterhalen op welke wijzen UML in de praktijk door software engineers gebruikt wordt. De resultaten hiervan geven aan dat software engineers de compleetheid en de mate van detail van UML modellen in het algemeen als vrij laag beoordelen. Bovendien geven software engineers aan dat incompleetheid van modellen die het ontwerp beschrijven een aanleiding is om in de implementatie fase af te wijken van het ontwerp. De invloed van incompleetheid is verder onderzocht aan de hand van een industriële case studie. Uit het vergelijken van de kwaliteit (gemeten in termen van fout-dichtheid) van componenten die wel met UML zijn gemodelleerd met de kwaliteit van componenten die niet met UML zijn gemodelleerd, bleek dat componenten die wel gemodelleerd waren een significant hogere kwaliteit hebben dan de componenten die niet gemodelleerd waren. Deze vinding suggereert dat de compleetheid van modellen van groot belang is voor het realiseren

van een hoge kwaliteit van de implementatie. Bovendien werd gevonden dat het gebruik van UML samenging met een kleinere inspanning voor het repareren van fouten. Deze resultaten werden gevonden terwijl rekening werd gehouden met mogelijke versturende factoren zoals koppelingsgraad en complexiteit van classes.

In de praktijk is een grote variatie gevonden in de mate van detail waarmee UML modellen worden gemaakt. De mate van detail (in Engels: Level of Detail, afgekort met LoD) representeert de hoeveelheid informatie die gebruikt wordt om een model te beschrijven. De invloed van LoD werd onderzocht door middel van een experiment waaraan 53 masterstudenten deelnamen. De uitkomsten van het experiment laten zien dat een hogere mate van LoD samenhangt met een beter begrip van het systeem en met het meer efficiënt (sneller) begrijpen van het systeem. Voor het meten van LoD is een aantal UML metrieken gedefinieerd. Deze metrieken zijn toegepast op een model uit een industrieel project. De uitkomsten hiervan laten zien dat (een hogere) LoD van UML modellen correleert met (een betere) kwaliteit van de implementatie. In het bijzonder bleek de LoD van sequence diagrammen een factor die van invloed is op de kwaliteit van de implementatie. Een aanvullende bevinding is dat UML metrieken gebruikt kunnen worden om de kans op aanwezigheid van fouten in implementatie-componenten te voorspellen: de metrieken LoD van messages en import coupling (IC) die gemeten worden op basis van sequence diagrammen zijn significante voorspellers voor de kans op fouten in de implementatie. Bovendien blijkt dat een voorspellingsmodel dat enkel op metrieken uit het UML ontwerp is gebaseerd een hogere nauwkeurigheid heeft dan een voorspellingsmodel dat enkel op code metrieken is gebaseerd.

Deze serie van empirische studies beoogt een bijdrage te geven aan de beantwoording van een centrale vraag omtrent de voor- en nadelen van modelleren met UML voor software ontwikkeling vooral vanuit een perspectief van software kwaliteit. De uitkomsten van deze empirische studies laten zien dat het gebruik van UML voordelen biedt in termen van toegenomen kwaliteit en productiviteit van software ontwikkeling. Het gebruik van UML modellen in vroege stadia van software ontwikkeling maakt het ook mogelijk om voorspellingen te doen over kans op fouten in software systemen. Zulke voorspellingen zijn potentieel bruikbaar voor het identificeren van foutgevoelige componenten en prioriteren van deze componenten voor testen.