وزارة التعليم العالي والبحث العلمي
جامعة الموصل
كلية علوم الحاسوب والرياضيات
قسم البرمجيات

تخمين حجم برامج جافا بالاعتماد على مخططات الصنف

رسالة مقدمة
الى مجلس كلية علوم الحاسوب والرياضيات في جامعة الموصل
كجزء من متطلبات نيل شهادة ماجستير علوم في
البرمجيات

من قبل

عائشة لقمان يحيى

بإشراف
أ.م.د. أسماء ياسين حمو

الخلاصة

يعد التقدير المبكر لحجم البرمجيات من المهام الصعبة التي تواجه مديري المشاريع ومهندسي البرمجيات لكون المعلومات المتاحة محدودة وهذا يؤدي الى نوع من عدم اليقين الحاصل نتيجة نقص او تغيير في المتطلبات إذ ان الخطأ في تقدير حجم البرمجيات يمكن ان يؤدي الى تقدير جهد المشروع بصورة خاطئة وبالتالي فشل المشروع. لذا فهي من اهم العمليات في هندسة البرمجيات. ولهذا فقد استخدم الباحثون ومديري المشاريع طرائق عديدة ومتنوعة ذات كفاءة عالية في تقدير حجم البرمجيات مثل خطوط الكود (LOC) ، وتحليل نقاط الوظيفة (FPA) ، مخططات (UML) ، وما إلى ذلك كون حجم البرنامج هو أحد العوامل الرئيسة لتحديد التكلفة والجهد والميزانية في عملية تطوير البرمجيات وتسليمها ضمن الجدولة المحددة لما يعود بالسمعة الحسنة لفريق البرمجيات.

من اجل الحصول على حجم اقرب مايكون للحجم الحقيقي فقد اخترنا في هذا العمل الاعتماد على مخططات (UML) في تخمين حجم البرمجيات وبالاخص مخطط الصنف في مرحلتي التحليل والتصميم لما يحتويه من معلومات هامة تمثل البرنامج فيما بعد.حيث تم اختيار اربعة مقاييس مختلفة لمخطط الصنف وهي عددالاصناف الكلية في المخطط وعددالسمات وعدد العمليات وعدد العلاقات التي تربط بين الاصناف ومن ثم تم استخدام هذه المقاييس الاربعة بمعادلتين يتم من خلالهما حساب عدد اسطر الشفرة البرمجية لكل مشروع.

تم تطبيق الطريقة على عشرة مخططات للصنف ستكتب لاحقا بلغة جافا وتم التاكد من صحة النموذج من خلال تطبيق النموذج على ثلاثة مشاريع حقيقية مكتوبة بلغة جافا ومن ثم عمل مقارنة بين الحجم الحقيقي للبرنامج والحجم المقدر وكانت النتائج ان الحجم المقدر قريب بنسبة 70% من الحجم الحقيقي.

تم بناء واجهة للنموذج ليتمكن مهندسي البرمجيات ومدراء المشاريع من استخدام النموذج بشكل صحيح. حيث يتطلب استخدام النموذج وجود مخططي صنف احدهما لمرحلة التحليل والاخر لمرحلة التصميم ثم تحويلهما الى مستندي XML ليتمكن النموذج من التعامل معهما.

Ministry of Higher Education and
Scientific Research
University of Mosul
College of Computer Science and
Mathematics
Department of Software

Estimation of java language codes size based on class diagram

A Thesis Submitted to the Council of the College of Computer Science and Mathematics University of Mosul as a Partial Fulfillment of Requirements for the Degree of Master of Science in Software

By

Aisha Luqman Yahya

Supervised by

Assist. Prof. Dr. Asmaa Yaseen Hamo

1445 A.H. **2023 A.C**

Abstract

Early estimation of the size of the software is one of the difficult tasks facing project managers and software engineers because the available information is limited and this leads to a kind of uncertainty resulting from a lack or change in requirements, as the error in estimating the size of the software can lead to a wrong estimate of the project effort and thus failure The project. Therefore, it is one of the most important processes in software engineering. Therefore, researchers and project managers have used many and varied methods with high efficiency in estimating the size of software such as lines of code (LOC), function point analysis (FPA), diagrams (UML), etc., since the size of the program is one of the main factors for determining the cost, effort and budget in Software development process and delivery within specified schedule to the goodwill of the software team.

In order to obtain a size as close as possible to the real size, we have chosen in this work to rely on (UML) diagrams in estimating the size of the software, especially the item diagram in the analysis and design phases, because it contains important information that represents the program later. Where four different metrics were chosen for the item diagram, namely The number of total items in the chart, the number of attributes, the number of operations, and the number of relationships that link the items, and then these four measures were used with two equations through which the number of lines of code for each project is calculated.

The method was applied to ten class diagrams that will be written later in the Java language, and the validity of the model was confirmed by applying the model to three real projects written in the Java language, and then a comparison was made between the real size of the program and the estimated size. The results were that the estimated size is close to 70% of the real size.

An interface to the model has been built so that software engineers and project managers can use the template properly. Where the use of the model requires the existence of two class diagrams, one for the analysis stage and the other for the design stage, then converting them into two XML documents so that the model can deal with them.