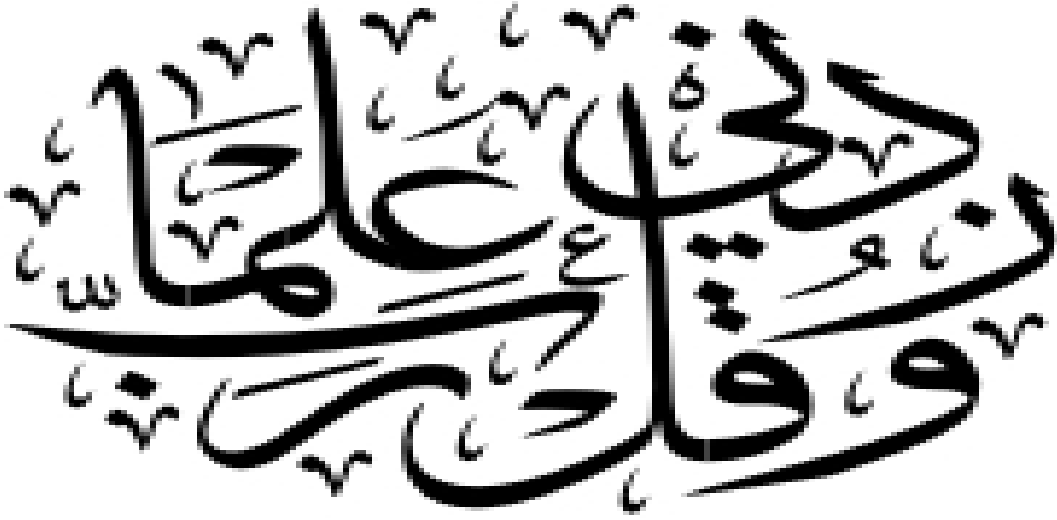


آية

بسم الله الرحمن الرحيم



صدق الله العظيم.
سورة طه ، آية (114).

إهداء

يقول تعالى " وَإِذْ تَأْتِيَنَّكُمْ رُبُّكُمْ لِيُنْزِلَنَّ عَلَيْكُمْ لِكُفْرَانِكُمْ أَنْزِيلًا كَثِيرًا لِيُذَكِّرَ الَّذِينَ كَفَرُوا إِنَّهُمْ كَانُوا فِي عَذَابٍ مُشْتَبِهٍ " وَإِذْ تَأْتِيَنَّكُمْ رُبُّكُمْ لِيُنْزِلَنَّ عَلَيْكُمْ لِكُفْرَانِكُمْ أَنْزِيلًا كَثِيرًا لِيُذَكِّرَ الَّذِينَ كَفَرُوا إِنَّهُمْ كَانُوا فِي عَذَابٍ مُشْتَبِهٍ ...

فله الحمد كما ينبغي لجلال وجهه وعظيم سلطانه وله الحمد والشكر على نعمه التي لا تعد وتحصى وبعد.

الى الوالدين الكريمين حفظهما الله ،،

الى إخوتي وأخواتي ،،

الى أهلي وأقاربي ،،

الى كل من له فضل في تربيتي و تعليمي ،،

الى كل هؤلاء اهدي ثمرة جهدي المتواضع ،،

شكر وتقدير

الحمد لله الذي وفقني لإنجاز هذا العمل وأنتهز هذه المناسبة لأتقدم بالشكر لجامعتي الغراء – جامعة السودان للعلوم والتكنولوجيا – على قبولها لي في هذا البرنامج متمثلة في كلية علوم الحاسوب ونظم المعلومات – الدراسات العليا، وأخص في ذلك أستاذي الفاضل الدكتور/ محمد عوض الشيخ المشرف على الرسالة، لما قدم لي من نصح وتوجيه ومتابعة مستمرة، رغم كل مشاغله وارتباطاته العلمية والإدارية المتعددة. فلقد كانت توجيهاته ذات أثر بالغ في نجاح هذه الدراسة وإتمامها. كما أن متابعته المستمرة وتوجيهاته السديدة وملاحظاته النافذة كانت العامل الأساسي في إخراج هذا العمل على النحو الذي ظهر به.

كما أتقدم بعظيم شكري وتقديري إلى المناقش الداخلي د. عوض محمد عوض الكريم لقبوله الاشتراك في لجنة مناقشة هذه الرسالة والحكم عليها، كما أشكره بعمق على تفضله بقراءة الرسالة، وهي في مسودتها الأولى، على الرغم من كثرة مشاغله وارتباطاته.

وشكر وتقدير لا حدود لهما للمناقش الخارجي د. اسامة الريس لتكرمه بقبول المشاركة في لجنة مناقشة هذه الرسالة والحكم عليها، على الرغم من كثرة مسؤولياته وضيق وقته.

كما أشكر أساتذتي الإجلاء وزملائي الأفاضل بقسم الحاسوب لما قدموه لي من نصح و عون صادق.

وفي الختام أشكر كل من قدم لي نصحاً أو مساعدةً أو توجيهاً وإرشاداً أثناء القيام بهذه الدراسة، فلهم مني خالص الدعاء، وللجميع التوفيق والسداد.

معتصم النور الطيب

المستخلص

تقدم هذه الأطروحة دراسة ووصف للأساليب التي يمكن استخدامها لتطوير نظام دعم أفضل لهندسة البرمجيات التعاونية والتي تتلخص في الاستفادة من عدة خبراء (محللين، مصممين، ومبرمجين ...) موجودون في بيئة موزعة إلى حد كبير. يسعى هؤلاء المهندسون لتطوير مشروع مشترك كل حسب تخصصه وخبراته وبالتالي الاستفادة من خبرات بعضهم البعض.

إن الهدف من هذا البحث هو إجراء دراسة مقارنة للمنهجيات التي توفر دعم للعمل التعاوني بشكل محدد - مقارنة هذه المنهجيات مع بعضها البعض - .

تم استعراض المنهجيات ذات الطابع التقليدي المركزي وتلك التي توفر دعم للعمل التعاوني (التوزيعية)، وكذلك تم استعراض الخصائص وأوجه القصور لكل منهجية على حده. كما تعرضت الأطروحة من خلال هذه المقارنة لبعض أوجه التشابه والاختلاف بين هذه المنهجيات، بالإضافة إلى شرح الإيجابيات والسلبيات لكل واحدة منها.

وتلقى هذه الدراسة أيضاً الضوء على الخطوط العريضة- من خلال إبراز الاستخدامات المناسبة- والتي يمكن أن تكون بمثابة تعاون مستقبلي بين المشاريع وخاصة في إجراء دراسات مقارنة بين هذه المنهجيات في الإطار العملي، وتعتبر هذه الخطوط العريضة هدفاً آخر لهذا البحث.

وقد ذكر الباحثون في مجال الأنظمة الموزعة أن منهجية البرمجيات التقليدية يمكن أن توفر فوائد كبيرة للمشاريع الصغيرة والمتوسطة، ولكنها تبدو قاصرة مع مشاريع ذات حجم كبير. وبينما لا تدعم هذه الأدوات التقليدية (والتي عادة ما تركز على دعم المشاريع الفردية)، الأنظمة التي يشارك فيها مهندسون ومطورون موجودون في عدة مواقع، وحسب ما ذكره الباحثون في هذا المجال، فإن منهجية جينيسيس يمكن أن تقدم دعم تعاوني لمشروعات توصف بأنها كبيرة و غير متجانسة.

يمكن أن يتحقق دعم أكبر إذا تم تطبيق هذه المنهجيات على نطاق واسع في الإطار العملي على خلفية الموجهات التي قدمتها الأطروحة.

ABSTRACT

This thesis introduces a study and description to methods that can be used to develop a better support system for collaborative software engineering, which is to take advantage of several experts (Analysts, designers, and programmers ...), who are in a distributed environment to a large extent. Those engineers seeking to develop a joint venture, each according to his specialization and experience and thus benefit from the experiences of each other.

The objective of this research is to conduct a comparative study for methodologies that provide support for collaborative work specifically - to compare these methodologies with each other - .

Traditional methodologies, which reflect the central work, were presented. Also the characteristics and limitations of each approach were introduced. Through this comparison, the thesis has been exposed to the similarities and differences between these methodologies, In addition to explaining the pros and cons of each one of them.

This study has also highlighted the broad outlines - by highlighting the appropriate use -, which can serve as future cooperation between the projects specially in doing a comparative study between these methodologies in a practical context, and these guidelines are considered as another objective of this research.

Researchers of collaborative Software engineering field have reported that, Traditional software paradigms can provide good benefits for small-scale and medium-scale projects, but for large-scaled projects traditional methods seem to fall short. As collaborative system has been poorly supported by traditional CASE tools (which usually concentrate on supporting individual projects), and according to what the researchers have mentioned in this field ,it was found that GENESIS provided support for maintaining large distributed collections of heterogeneous software components across a number of projects.

The long-term benefits of such support can only be realized if these approaches were widely adopted according to the described outlines.

LIST OF ABBREVIATIONS

CRC	Class Responsibility Collaborator.
UML	Unified Modelling Language.
ORM	Object-Role Modeling.
AM	Agile Modelling.
LDM	logical data model.
PDM	physical data model.
XP	eXtreme Programming.
RUP	Rational Unified Process.
DBMS	Database Management System.
FMC	Fundamental Modelling Concepts.
FEDEP	Federation development and execution process.
CSCW	Computer Support for Collaborative Work.
SDM	Software Development Methodology.
GENESIS	Generalized Environment for procESs management in cooperative Software engineering.
CSCW	Computer Supported for Collaborative Work.
DSE	Distributed Software Engineering.
CSE	Co-operative Software Engineering.
MAS	Multi-Agent Systems.

MIS	Module Interface Specifications .
DiSECT	Distributed Simulation Exercise Construction Toolset .
PSEE	Process-sensitive Software Engineering Environmen
HTTP	Hypertext Transfer Protocol
ERPs	Enterprise Resource Planning systems
DRPs	Web-based Development Resource Portals
SCM	Software Configuration Mgmt.
OPHELIA	Open Platform and meTHodologies for deVELopment tools and IntegrAtion in a distributed environmen.

فهرسة الأشكال

الصفحة	إسم الشكل
10	شكل 1-2 عملية نمذجة البيانات.....
16	شكل 2-2 مثال لمخطط الهيكل التركيبي.....
20	شكل 3-2 أنواع رسومات لغة النمذجة الموحدة.....
42	شكل 1-4 مثال لموقع جينيسيس GENESIS.....

الفهرسة

الصفحة	الموضوع
i	آية.....
ii	الاهداء.....
iii	شكر وتقدير.....
v	المستخلص.....
viii	الإختصارات.....
x	فهرسة الأشكال.....
xi	الفهرسة.....

الباب الأول: مقدمة

1	1.1 تمهيد.....
3	2.1 تعريف المشكلة.....
3	3.1 أهداف البحث.....
4	4.1 منهجية البحث.....
4	5.1 حدود البحث.....
4	6.1 تنظيم البحث.....

الباب الثاني: خلفية تاريخية

5	1.2 مقدمة.....
---	----------------

6	2.2 دورة حياة نموذج البرمجيات.....
8	3.2 نموذج عملية البرمجيات.....
9	4.2 دورة حياة نماذج البرمجيات التقليدية.....
10	5.2 عملية نمذجة البيانات.....
11	1.5.2 نمذجة البيانات.....
11	1.1.5.2 نمذجة البيانات الإستراتيجية.....
11	2.1.5.2 نمذجة البيانات اثناء تحليل النظم.....
13	2.5.2 العمليات التجارية.....
14	3.5.2 منهجيات النمذجة.....
15	4.5.2 أنواع المخططات.....
16	1.4.5.2 مخطط البناء التكويني.....
17	2.4.5.2 مخطط البناء الديناميكي.....
17	3.4.5.2 مخطط البناء ذو المدى القيمي.....
18	6.2 لغة النمذجة الموحدة.....
20	7.2 النمذجة اللامركزية والمستقلة.....
22	1.7.2 اللامركزية.....
23	2.7.2 اللامركزية مقابل التوزيع.....
24	8.2 منهجية النمذجة اللامركزية.....
الباب الثالث: تطور هندسة البرمجيات الموزعة	
26	1.3 مقدمة.....
26	2.3 الأنظمة الداعمة للعملية.....
27	3.3 منهجية ريدي.....

27	4.3 مقترحات مشروع انديفرس.....
27	5.3 نظام تينق و مورير.....
28	6.3 نظام جراندي وآخرون.....
28	7.3 بيئة اوزويب.....
28	8.3 نمذجة المتطلبات الحقيقية لتطوير البرمجيات التعاونية.....
29	9.3 المعمارية القائمة على العميل.....
30	10.3 إدارة المكونة المفتوحة المصدر.....
31	11.3 دعم مجموعات كبيرة من وحدات البرمجيات.....
31	12.3 خلاصة.....

الباب الرابع: نبذة عن المنهجيات

32	1.4 مقدمة.....
32	2.4 إدارة تطوير البرمجيات الموزعة.....
33	1.2.4 البقاء على إتصال.....
33	1.1.2.4 الإتصالات هي المفتاح.....
33	1.1.1.2.4 المناقشات المتشابكة (المترابطة).....
33	2.1.1.2.4 التراسل الفوري.....
34	3-1-1-2-4 المدونات.....
34	4-1-1-2-4 المؤتمرات علي الشبكة.....
35	2.1.2.4 المستودعات التشاركية.....
36	3.1.2.4 اهمية مشاركة المطورين.....
36	1.3.1.2.4 العروض التقديمية عبر الإنترنت.....
36	2.3.1.2.4 المراجعات.....

363.3.1.2.4 الزيارات الميدانية.
373.4 معمارية العملية التقليدية الداعمة لهندسة البرمجيات التعاونية.
384.4 المعمارية القائمة على الوكلاء المتعدون لـ CSE.
381.4.4 الوكلاء.
392.4.4 فضاءات العمل.
393.4.4 أقوراس.
404.4.4 المستودعات.
405.4 تمهيد عن جينيسيس.
426.4 تمهيد عن أوفيليا.

الباب الخامس: دراسة مقارنة

441.5 مقدمة.
442.5 أنماط أو منهجيات البرمجيات التقليدية.
441.2.5 الإيجابيات.
442.2.5 العيوب أو السلبيات.
453.2.5 الإستخدام (المواقع المناسبة).
453.5 الأنماط أو المنهجيات القائمة علي تعدد الوكلاء.
451.3.5 الإيجابيات.
462.3.5 العيوب أو السلبيات.
473.3.5 الإستخدام (المواقع المناسبة).
474.5 جينيسيس و اوفيليا.
471.4.5 الإختلافات الأساسية وأوجه التشابه.
492.4.5 الإيجابيات.

49 3.4.5 العيوب أو السليبيات.
50 4.4.5 الإستخدام (المواقع المناسبة).
51 5.5 مقارنات أخرى.
52 6.5 خطوط عريضة.
52 7.5 إعتبارات.

الباب السادس: الخاتمة والعمل المقبل

54 1.6 خاتمة.
55 2.6 التوصيات والأبحاث المستقبلية.
56 المراجع.