**Предложение**

**за**

възлагане на дипломна работа

към катедра “Софтуерни технологии”,

ФМИ, СУ “Св. Климент Охридски”

**От дипломант:** Димитър Манев , специалност СТ, факултетен № М-22499

**Научен ръководител:** доц. Александър Димов, Софтуерни Технологии/ ФМИ, СУ “Св. Климент Охридски”

**Консултант:** ....................................................., катедра/ВУЗ/институт .........................

**Тема на дипломната работа:**

**“Mоделно базирана разработка на софтуер за вградена софтуерна система чрез автоматично извличане на архитектурна информация”**

**Анотация:**

Вградените системи продължават да навлизат все повече и повече в нашия живот, като растежа на разпространението им е огромен. Тези системи трябва да отговарят на нарастващ брой изисквания за функционалност, време за реагиране, ограничения към процесорно време и памет, консумация на енергия, цена и т.н. Същевременно стандартните подходи за компонентно базирана разработка на софтуерното инженерство като CORBA, DCOM/COM, Enterprise JavaBeans и т.н. не са подходящи за вградените системи, тъй като компонентите създадени от тях, не отговарят на голяма част от гореспоменатите изисквания освен това тъй като най-често езика за разработка е „C” става почти невъзможно да се прилагат и възможностите на инструментите за моделиране, особено извличане на моделна информация и генериране на код.За да могат да бъдат конкурентно способни тези системи трябва да могат лесно да се прекрояват. Това налага разширяване на средата за дизайн на такива системи, с нови автоматизирани инструменти даващи възможност голям брой от изискванията към системата да бъдат проверени в ранен етап на разработката. Решение за това е използването на специфични компоненти за областта на приложение в съчетание със специфичен модел на разработваната система.

Възможността за извличането на архитектурна информация от изходния код на вградени системи ще даде възможност за автоматизиран анализ на свойствата на тази система, нещо което най-често се прави на ръка и дава възможност за пропуски, съответно и до дефекти на самата система. От друга страна реконструирането на архитектура от модел извлечен от вече валидирана и успешно посрещнала качествените си изисквания система, допринася до значително намаляване на дефектите на бъдещата система. Удобно би било да се създаде приложение за анализиране на вече съществуващи системи, което изготвя модел отговарящ на компонентите и интерфейса, по който си комуникират един с друг както и информация за системата в динамичен вид. По този начин се извлича информация за архитектурата на вече създадени системи, което значително улеснява дизайна на нови системи с подобни изисквания. Тъй като в момента UML (Unified Modeling Language) е стандартния за индустрията език за моделиране, за който съществуват множество инструменти и формати за представяне, удачно е той да бъде избран за представяне на архитектурните модели. Както и възползвайки се от предимствата на UML, ще се ускори разработката на нови системи и ще се подобри документацията и поддръжката както на системите в разработка, така и на тези, които са вече в продукция. Предизвикателството на създаването на такъв инструмент, е че повечето вградени приложения са разработени и продължават да се разработват на езикa за програмиране „C”. Това налага нестандартен подход към анализа на тези системи, тъй като за разлика от съвременните обектни езици за програмиране(като C++, C#, JAVA и т.н), езика „C” е слабо поддържан от средствата за UML обработка и дизайн.

**Цел на дипломната работа:**

Да се разработи архитектурен инструмент, който извлича информация за интерфейсите на компонентите от техния код и представя еквивалентен UML модел със следните артефакти: класове, компоненти(зависимости м/у компоненти). Инструментът трябва да може да генерира базов код за нова система по даден модел.

**Задачи, произтичащи от целта:**

1. Да се направи проучване и анализ на други подобни инструменти и подходи
2. Да се изследват отворени формати за представяне на UML модели и да се избере най-подходящия за извършване на задачата.
3. Да се създаде проект на софтуерния инструмент за генерация на базов код на системата, които съдържа три основни модула: модул за генериране на UML, анализатор на софтуерен код, генератор на код.
4. Да се разработи модул за генериране на UML модели.
5. Да се разработи анализатор на софтуерен код.
6. Да се разработи генератор на базов код на системата по даден модел.

**Ограничаващи/облекчаващи условия:**

няма

**Срок за изпълнение:** 2015-02-15

Дата: 2015-01-04 Заявител:

/студент/

/научен р-л/