Hacettepe University Department of Computer Engineering
Spring 2009-2010
BİL342 Software Laboratory II
Experiment 2

Name and Surname : İBRAHİM ŞAHİN
Identity Number : 20673618
e-mail : ibrhmsahn@gmail.com

Advisors: Prof.Dr. Ersin Töreci, R.A. Ersin Er
Subject: Domain Specific Formal Language Processing
Environment: Java 1.6, Antlr 3.2, JUnit 4.5, Maven 2.2
Date due: 24.03.2010 - 23:55

1.Software Design Notes

1.1. Software Using Documentation

You can think this project like a new language.We think this experiments is simple model of the checking submit system but you can use with different aims.

Software is compare the specific code with given zip file.In the code documentations attributes (mandatory,subject ,date etc) are basic elements of the code.This software check the zip file according the writen code.To use this program with tests is useful.

1.2. Problem and Solution

In this experiment we are desining a specific domain language.A submit system checking model have wanted from us.We must define firstly basic description (integer,string etc).In the format other improvement descriptions are subject,advisors,enddate,startdate,submitformat.We are using string or integer or other subspecies.

We are defining these descriptions in antlr and for some controls and hold the datas we are using java codes.There are MySubmitModel class.After the definition we are setting all variables to model.So we carry datas.If there is some definition characters system is not work.

I follow the flow of the program from the skeleton.In skeleton neccessary class have given.In additional the skeleton I added MySubmitFormat class.

2. Main Data Structures

```
public class MySubmitFormat {
      public int numero;
      public String element_name;
      public String[] element_order = new String[2];
      public int item;
```

In MySubmitFormat class these attributes are existing.I do not reprobate the directory,file,filepattern classes seperataly.I am differentiate these values with numero attributes of the MySubmitFormat object.I seperate the arraylists in the checkSubmission.

In the advisors I defined an arraylist for them.Like the submitformat I get them from mysubmitmodel class.These arraylists's definition in the SubmitDsl.g.Definition of the model in here also.I setted all variables with model object ,without submitformat.

```
@members {
      List<String> arrayList = new ArrayList<String>();
      MySubmitModel model = new MySubmitModel();
      List<MySubmitFormat> bigList = new ArrayList<MySubmitFormat>();

}
```