Boğaziçi University
Computer Engineering Department

CMPE160 project

CONTENT DEVELOPMENT TOOL

Author:

Halil Cetiner

Yusuf Hakan Kalayci

Supervisor:

Assoc. Prof. Haluk Bingol

May 2015

Abstract

Purpose of the Content Development Tool project is to prepare slides and presentation which stores information in Html5 format by using simple user interface. It is a desktop application developed with JavaFx.

Basically the Content Development Tool is used to prepare presentations with multiple slides. Mainly it is developed for programming lectures. We also added code displayers and options switch to code.

Following a period of rapid development since March of 2015, we had several discussions on this project with our professor and our research suggested that this project can help people who uses slides for their programming lectures.

Acknowledgements

Contents

Introduction

The Content Development Tool is a Bogazici University Computer Engineering Department project that aims to create slides which use some properties of Html5 format such as convenience of modification and flexibility between different platforms. We can create presentations with this tool that can be used in websites, eclipse plugins.

The tool is developed with JavaFx and it uses some additional data such as html files and css files. This addtional data provide creating default pages or adding additional facilities and setting up default view.

There are some projects which are similar to ours but they are commonly developed for html and as browser application. This project was developed as desktop application. And also this project can be modified and developed more.

Desing and Implementation

1.Setting up Environment

As we mentioned, this project is JavaFx project then you must setup eclipse for javafx. You can adjust your eclipse by using this link(http://www.eclipse.org/efxclipse/install.html) or you can also download all-in-one ide which is e(fx)clipse from that link.

After installing ide, you can import our project to your workspace.

2.User Interface

a) Menu

In this menu we created some menu items which provides quick access some operations. These operations can be listed as file operations, editor operations.

b) List

In this section we are listing slides which are in same project. Users can reach slides by using List.

It has 2 operations which edit slide list: adding slide and removing slide.

Figure 1

c) Extended Html Editor

In this section users can edit current slide. This section has many buttons and a Html5 viewer for displaying current view.

i) Html5 Viewer

Users can edit current view of slide by using Html5 viewer. They can see last state and they can write on it or they can add some special fragments by using buttons.
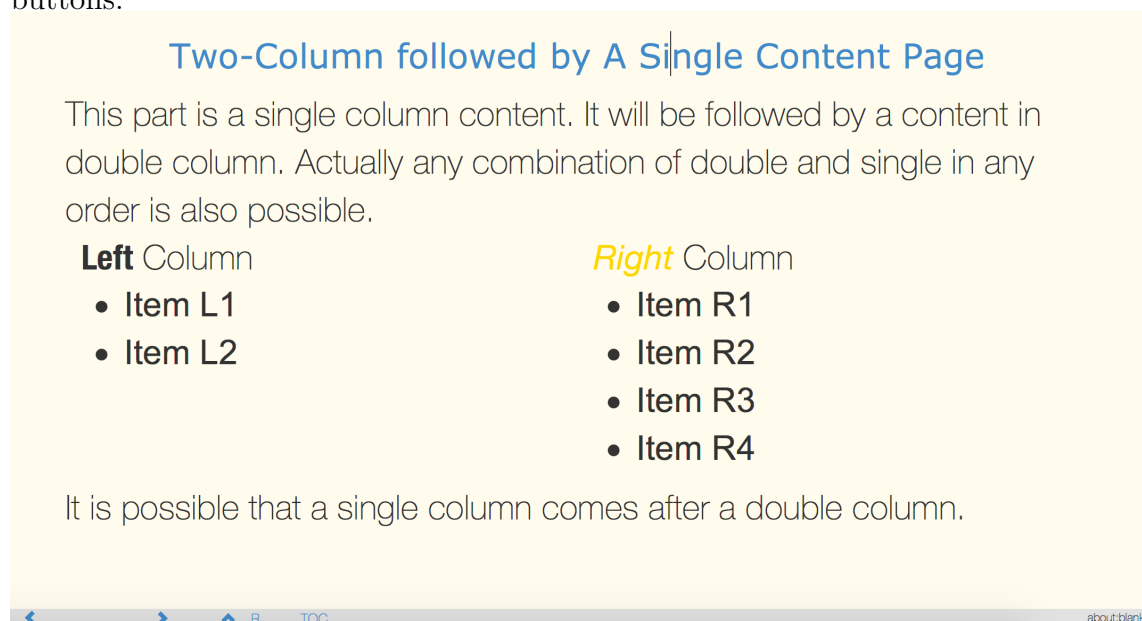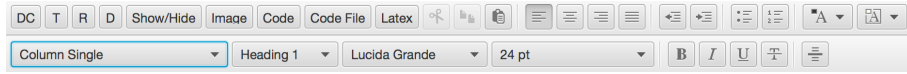

Figure 2

ii) Buttons

Figure 3

We have some buttons which are gained from HtmlEditor in JavaFx. These buttons are used for basic operations such as changing font size and type, aligning text, indentation, listing and etc.

We have also some buttons which are created by us. They can be listed such as:

DC (Double Column): This button adds double column at cursor position.

T (Theorem): This button adds theorem fragment which can be used for specify theorems.

R (Remark): This button adds a fragment which can be used for remarking some points.

D (Definition): This button adds definition fragment which can be used for specify definitions.

Show/Hide: This button adds a area which can be hidden and shown.

Image: This button displays a pop-up window and using this window users can add images to slides. These images can be added from both internet and local files.

Code: This button displays a pop-up windows and using this window users can write codes to slides. While displaying slide this code fragment beautify codes.

Code File: This button displays a pop-up window and using this window users can add switch to code button to their slides. This switch to code button provides quick access to ide.

Latex: This button displays a pop-up window and using this window users can add mathmatical terms and expressions easily in latex format.

Layout Chooser: With this JavaFx combobox users can change layout of slide.

On the other hand, we are using similar processes to apply. In these processes

5

we have some html code fragments for each button we are adding this fragments to html text of current slide. While adding fragment we are executing javascript codes in Html5 Viewer. These parts are handled in HTMLCodeFragmentInserter class.

3.Background

a) Basic Html Operations

In this part we implemented a HTMLReader class which reads html codes.

On the other hand we are using method which works with parse-edit-merge logic. Let us explain parse-edit-merge logic:

When we looked slides in html files we faced with a default layout. This layout can be explained in 5 parts. These parts can be seen at figure.
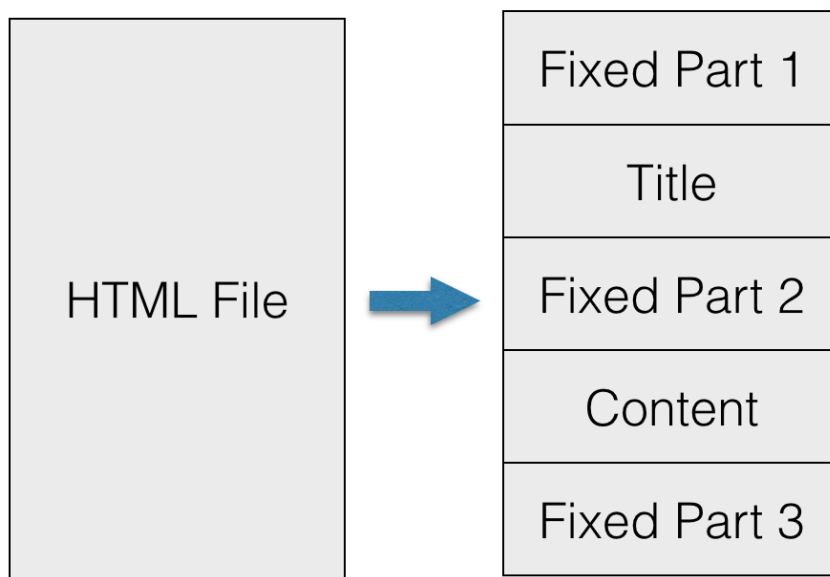

Figure 4

This tool does not edit fixed parts because our slides has identical theme. However by using this tool we can edit containers which are named title and content. We are handling this operation with HTMLParser and HTMLMerger classes. HTMLParser class detects editable parts of html file and parses them. However, HTMLMerger class merges editable parts and fixed parts in predefined order which is shown in figure.

b) History

In this part we will explain history operations such as undo and redo.

We are keeping some specified actions of users on the html file then providing them to use undo and redo.

c) Garbage Cleaner

JavaFx HtmlEditor adds some unnecessary codes to html file of slide, and our javascript files are adding some code to html file of slide at loading. While we are editting these garbages may cause some problems such as multiple "Theorem" or "Remark" expressions so we must clean up them. By using garbage cleaner we are cleaning up these garbages.

d) Image Download

Some slides can have images. We decided to copy images which are added to slide in fixed directory such /images or directory with name of slide. We are getting images to this directory by using this part.

e) Save

Our project provides users two different save options:

    i) Save singular slide: This option saves just current slide with a name which is consisted of current date and time to "bin/WebContent/SubjectA/SavedFiles".

    ii) Save whole project: This option saves whole project with a name which is consisted of current date and time to "bin/WebContent/SubjectA/SavedFiles".

4.Conclusion

In conclusion we have done fundemental part of the Content Development Tool. We developed Extended Html Editor which has special abilities to creating programming lectures' slides. We also prepared javaWe developed parts of project to provide convenience to maintenance and furthermore development. Especially we implemented html package and CodeFragmentInserter class considering this concept of convenience.

In our opinions, this project can be developed furthermore in many aspects. Firstly there can be a html code displayer to edit manually slide (without user interface), there can be minimized view on list of slides.

On the background, there may be problems with handling some unexpected behaviors, these kinds of behaviors can be handled more effectively.

Figures

Figure 1: Slide List

    Design and Implementation.2.b

    Page 4

Figure 2: HTML5 Viewer with 'Two-Column Followed by A Single Content Page' Layout

    Design and Implementation.2.c.i

    Page 4

Figure 3: Buttons

    Design and Implementation.2.c.ii

    Page 5

Figure 4: Default Layout Contents for Explain Parse-Edit-Merge Logic

    Design and Implementation.3.a

    Page 6

How to Access Project

Whole project can be downloaded from https://github.com/t0nyukuk/sd4u