

Assignment#6

DD-2

SE

Team:

Abay Serikov
Dina Makhmet
Merey Bolat
Gulbanu Kashkymbaeva

Team number: #7 “helloWorld!”

Design Document for Translit.kz

Author: Group 7: helloWorld!

Abay Serikov

Dina Makhmet

Gulbanu Kashkymbayeva

Merei Bolat

Version	Date	Author	Change
0.1	21/03/15	Group 7	Initial Design Document
0.2	04/04/15	Group7	Updated, with focus on 5.1, 5.2 section

Table of Contents

1	Introduction.....	3
1.1	Purpose	3
1.2	Scope	3
1.3	Definitions, Acronyms, Abbreviations	3
1.4	Design Goals	3
2	References.....	4
3	Decomposition Description.....	5
3.1	Module Decomposition	5
3.2	Concurrent Process	6
3.3	Data Decomposition	8
3.4	STATES	8
4	Dependency Description.....	9
4.1	Intermodule Dependencies	9
4.2	InterProcess Dependencies	9
4.3	Data Dependencies	9
5	Interface Description	10
5.1	Module Interface.....	10
5.2	Process Interface.....	12
6	Detailed Design.....	14
7	Design Rationale.....	15
7.1	Design Issues	15
7.2	<Issue 1>	15
7.3	<Issue 1>	15
8	Traceability.....	17

1 Introduction

<Make sure you follow guidelines in the ASSIGNMENT – as well as in this document>

<DO NOT DELETE ANY SECTION. INSTEAD, if you have nothing to write in any section, then write [NONE] or NA (not applicable) >

<< Please delete ALL instructor comments before submission>>

1.1 PURPOSE

Purpose of this document is to explain the design and architecture of the Translit.kz.

1.2 SCOPE

This document covers system decomposition, interfaces, and dependencies, as well as design rationale.

1.3 DEFINITIONS, ACRONYMS, ABBREVIATIONS

Term	Description

1.4 DESIGN GOALS (MB)

1. Reliability: The core process (major use-cases) must continually function consistently.
2. Maintainability: Our code must be well organized with consistent syntax and relevant naming techniques.
3. Extensibility: The application must facilitate easy extension and features to be added to the PHP Code.
4. Response Time: All Web pages must load in less than 30 seconds including library.

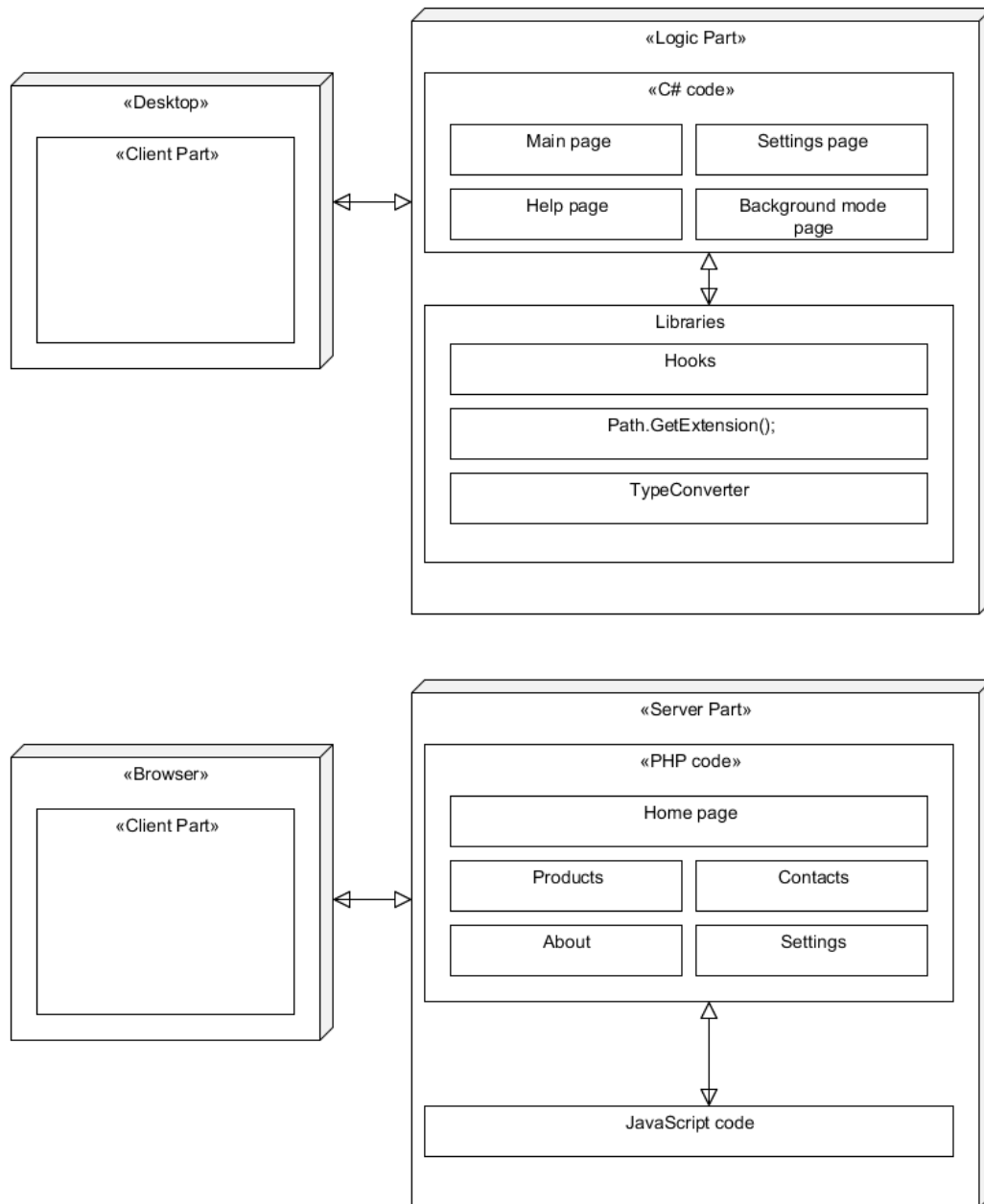
2 References

(If any)

3 Decomposition Description

3.1 MODULE DECOMPOSITION

ARCH DIAGRAM FOR TRANSLIT.KZ SYSTEM



These two diagrams have the same logic of construction Architecture Diagram. The first diagram describes the architecture of a desktop application. It consists of two layers. The first layer is code C# which includes all pages of the application. The second contains only the Libraries module, which provides services to the C# Code in the layer above. The second diagram describes the architecture of a web application. It consists of two layers. The first layer contains only PHP code, the second layer is JavaScript code, which provides services to the PHP code in the layer above. (MB)

3.1.1 C# Code Description(SA)

C# code is consisted with 4 different pages. Each page takes charge of specific events and graphical process. It uses libraries to get/send information.

3.1.2 Libraries Description(MD)

The libraries modules contains the classes and functions needed by the C# Code.

3.1.3 PHP Code Description (BM)

PHP code is consisted with 5 different pages. Each page takes charge of specific events and graphical process. It uses JavaScript to get/send functions.

3.1.4 JavaScript Description. (KG)

JavaScript modules contains the classes and functions needed by the PHP.

3.2 CONCURRENT PROCESS

<Give an overall description of the different processes, threads, instantiations>

3.2.1 Main Page Description (Desktop AS)

It is the main page of our desktop applications. There will be two TextBoxes. In the first TextBox you can type the text in Latin/Kirill and in another one get translited text by clicking on the button “Translit It”. Settings Page Description (Desktop AS)

3.2.2 Help Page Description (Desktop MD)

Page HELP teaches us how to change the settings and use the program.

3.2.3 Background Mode Page Description (Desktop MD)

On this page will be a CheckBox to enable or disable automatic background transliteration. If CheckBox Checked background mode is turn on, else turn off.

Hooks: A hook is a point in the system message-handling mechanism where an application can install a subroutine to monitor the message traffic in the system and process certain types of messages before they reach the target window procedure.

3.2.4 Settings page Description (Desktop MD)

In the Settings page, we can change the letters as we used to write. For example, someone wrote the letter III like W or someone like SH. Clicking on "Save changes " we keep the settings, or you can proceed without saving.

3.2.5 Home page Description (Web BM)

It is the main page of our desktop applications. There will be two TextBoxes. In the first TextBox you can type the text in Latin/Kirill and in another one get translited text by clicking on the button "Translit It". Also on this page you can get translited file by downloading it. We use the library of PHP method typeConverter.

TypeConverter: This method converts file from one extension to another. It needs for us, because when we translited file, firstly system must convert our extension to standard .txt or .doc etc.

Path.GetExtension: Returns the extension of the specified path string.

3.2.5.1. Settings Page Description

In the Settings page, we can change the letters as we used to write. For example, someone wrote the letter III like W or someone like SH. Clicking on "Save changes " we keep the settings, or you can proceed without saving.

3.2.6 Products Page Description (Web BM)

Our application will be developed and all of the latest updates of our desktop applications, we will throw on this page.

3.2.7 About Page Description (Web KG)

On this page we describe why we need the program, the audience of program etc.

3.2.8 Help Page Description (Web KG)

Page HELP teaches us how to change the settings and use the program.

3.3 DATA DECOMPOSITION

<Class 1> Description

<Class 2> Description

3.4 STATES

<State/System 1 > Description

<State/System 2> Description

4 Dependency Description

4.1 INTERMODULE DEPENDENCIES

4.2 INTERPROCESS DEPENDENCIES

4.3 DATA DEPENDENCIES

5 Interface Description

5.1 MODULE INTERFACE

5.1.1 JavaScript Code Interface(AS)

5.1.1.1 Function call which called by HTML (PHP) and provided by JavaScript

5.1.1.1.1 `function strtr(str, repl);`

used for transliteration each char value of string word in window(input text)

Parameters: str = original string taken from window;

repl = array from which we should take transliteration char value;

Return: str = translited string text;

5.1.1.1.2 `function replace(new RegExp(f, t), r);`

it's Regular Expression that should replace every char by another

Parameters: f = old value of char;

t = new value of char;

r = string that should be replaced;

Return: replaced value of string;

5.1.1.1.3 `function getTranslit();`

used for calling the transliteration method when text typed into window or when button

“TRANSLIT IT” clicked;

Parameters: no parameters;

Return: none;

5.1.1.1.4 `function handleFileSelect(evt);`

used for selecting file in dialog window from your computer when “Select file” button is clicked, it work by using standard element `<input type = "file">`. JavaScript returns a list of the selected objects in a File object `FileList`

Parameters: evt = var file which selected from dialog window;

Return: list of selected objects(files);

5.1.1.1.5 `function document.getElementById(ID);`

use for define objects in JavaScript which created in HTML(PHP) by using Identification value

Parameters: ID = identifier of object;

Return: none;

5.1.2 C# Code Interface (MD)

5.1.2.1 Dictionary library.

We use this library for transliterate Cyrillic to Latin and vice versa.

```
Dictionary<string, string> CyrtLat = new Dictionary<string, string>();
```

```
Dictionary<string, string> LattoCyr = new Dictionary<string, string>();
```

```
words.Add("а", "a");  
words.Add("б", "b");  
words.Add("в", "v");  
words.Add("г", "g");  
words.Add("д", "d");  
words.Add("е", "e");  
words.Add("ё", "yo");.../here we add chars by default to our Dictionary.
```

5.1.2.2 Hooks - we need it, so that we could use transliteration for all open windows.

```
public class KeyHook  
{  
    protected static int Hook;  
    protected static LowLevelKeyboardDelegate Delegate;  
    protected static readonly object Lock = new object();  
    protected static bool IsRegistered = false; }  
}
```

We created class KeyHook, after that called this class in our main code Translit.kz

```
KeyHook gHook;
```

After that we use Dictionary to realize Transliteration on each textbox.

5.1.2.3 path.GetExtension - Returns the extension of the specified path string. We need this for Convert from one type to another.

```
string fileName = @"C:\mydir\myfile.doc";
string path = @"C:\mydir.txt\";
string extension;

extension = Path.GetExtension(fileName);

extension = Path.GetExtension(path);

// This code produces output similar to the following:
//
// GetExtension('C:\mydir\myfile.doc) returns '.ext'
// GetExtension('C:\mydir.txt\') returns "
```

5.2 PROCESS INTERFACE

5.2.1 Main process (KG)

This process shows all graphical interface of the system

5.2.1.1. Process is created when the page of web application is opened

5.2.1.2. Terminated when the page of web applications close button is pressed

5.2.1.3 All other threads will be killed if this main thread stops

5.2.2 "Translit" process

This process shows transliting from kirill to latin or from latin to kirill

5.2.2.1 Process is worked when we enter the letters and press the button "translit"

5.2.2.2 Terminated when we remove the letters

5.2.3 "Settings" process

This process shows settings of the application, in which we change latin letters for yourself

5.2.3.1 Process is started when we change the latin letters of those we have or we write the letter which we want

5.2.3.2 Process is worked after the changes we press a button "save"

5.2.3.3 Lost if we don't save

5.2.3.4 Terminated when process is closed

5.2.4 "About" process (BM)

This process shows the description of service

5.2.4.1 Process is started when we press button "about"

5.2.4.2 Terminated when button "close" is pressed

5.2.5 "Select file" process

This process shows the selection the .doc or .pdf file and translit texts in those files

5.2.5.1 Process is started when we press the button "select file" and choose the file

5.2.5.2 Process ends when we get the transliting text

6 Detailed Design

NOT REQUIRED <Java Docs to be used instead>

7 Design Rationale

7.1 DESIGN ISSUES

7.2 <ISSUE 1>

Description

Factors affecting Issue

Alternatives and their pros and cons

Resolution of Issue

7.3 <ISSUE 1>

Description

Factors affecting Issue

Alternatives and their pros and cons

Resolution of Issue

8 Traceability

No	Use Case/ Non-functional Description	Subsystem/Module/classes that handles it
1		
2		

FEEL FREE TO ADD APPENDICES AS NEEDED. UPDATE TOC BEFORE SUBMITTING