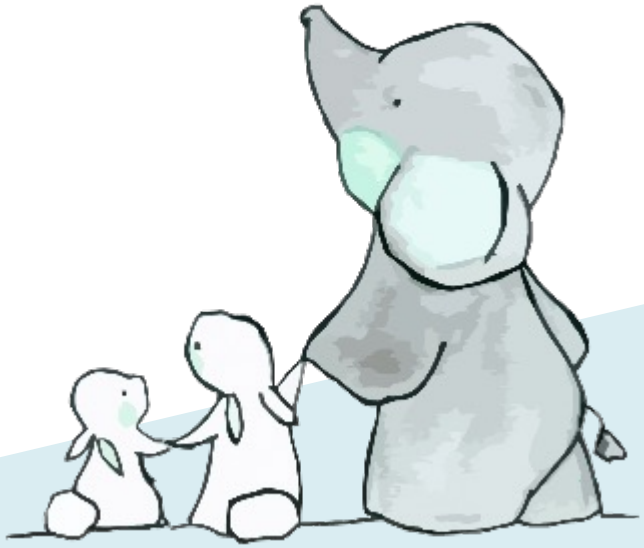
Software projects

Done by:Mariyam Mukhumova

Introduction

1. Education
2. Software projects on B.Sc and M.Sc
3. Software projects during professional experience
4. References

01 Education



Education

- **Bachelor degree (Almaty city, Kazakhstan)** :

Kazakhstan State Educational Grant & Scholarship (2011 - 2015)

Major: Information Systems

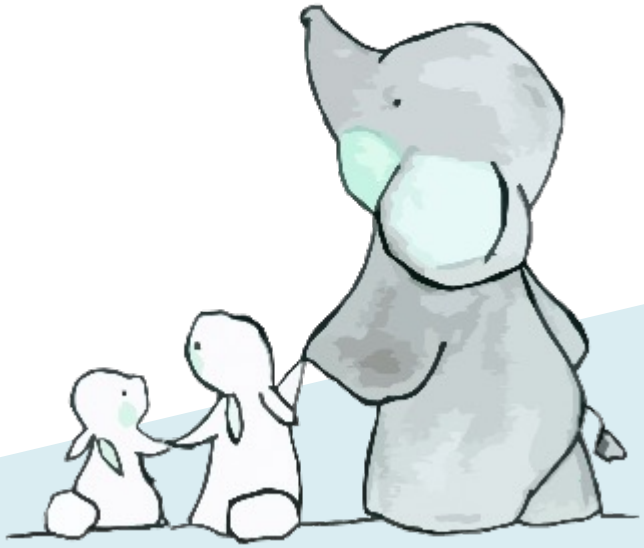
- **Master degree (Hangzhou city, China) :**

Zhejiang Sci – Tech University President Scholarship (2017 - 2020)

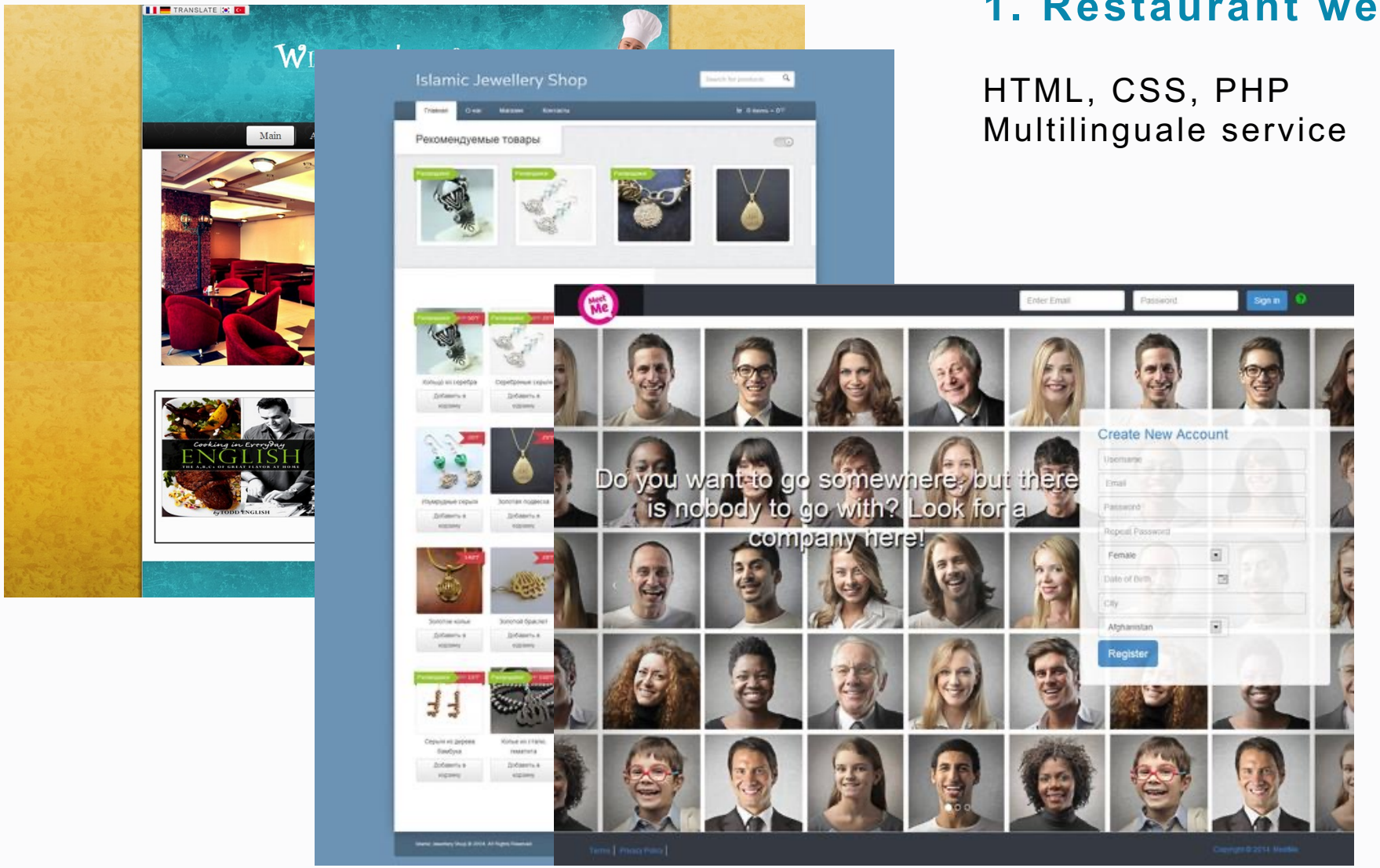
Major: Computer Applied Technology

02 Software projects

on B.Sc and M.Sc



Project #1



1. Restaurant website

HTML, CSS, PHP
Multilinguale service

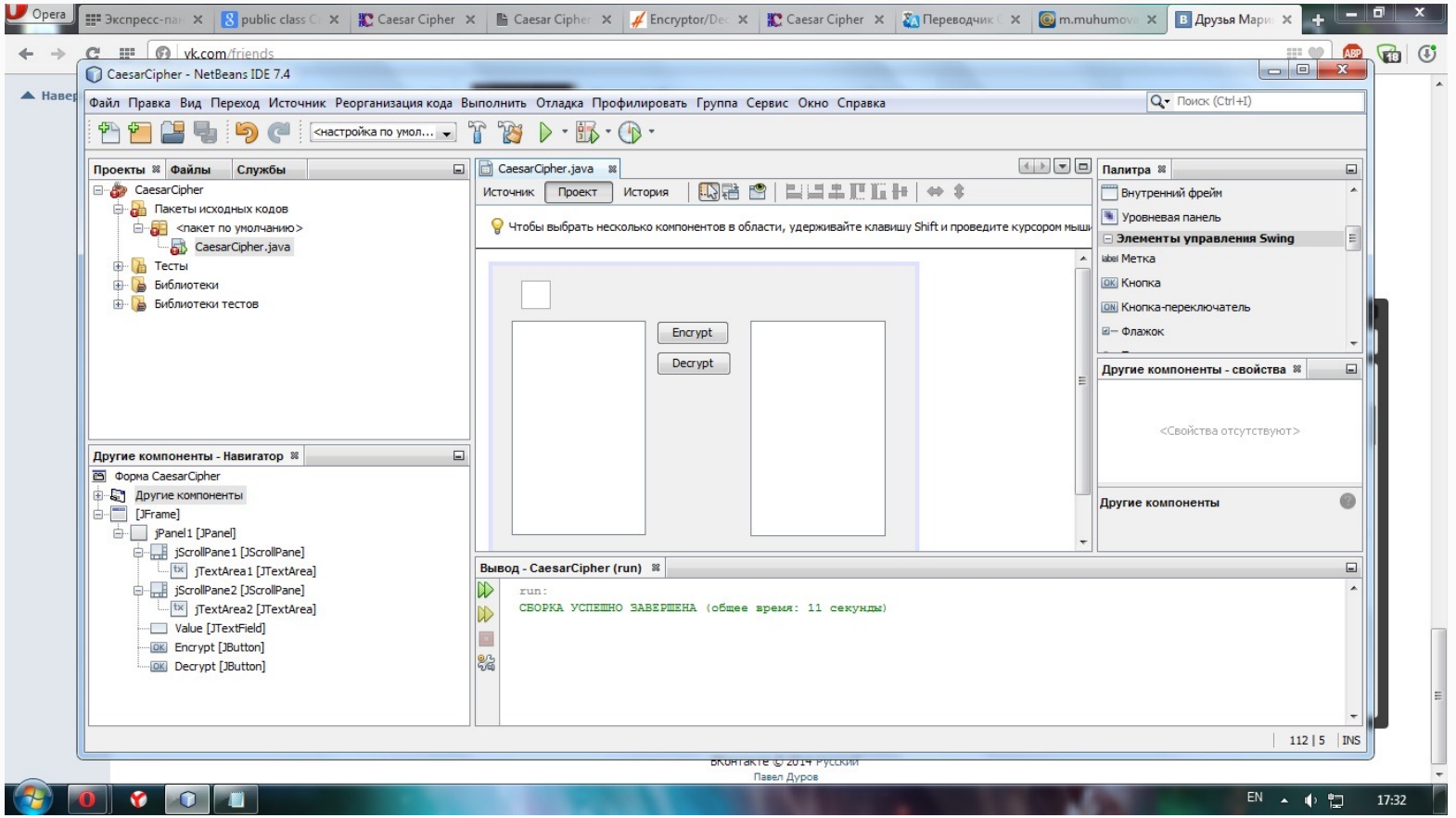
2. Online shop

CMS Wordpress
HTML, CSS, PHP
Woocommerce

3. Social network

PHP Framework
HTML, CSS, PHP
Writing Posts, Likes
Registration of profiles

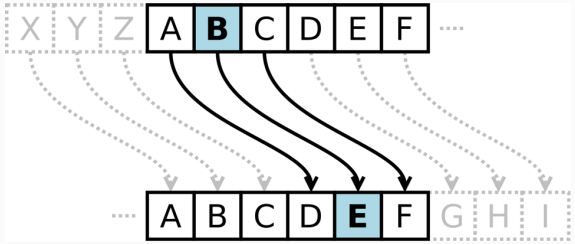
Project #2



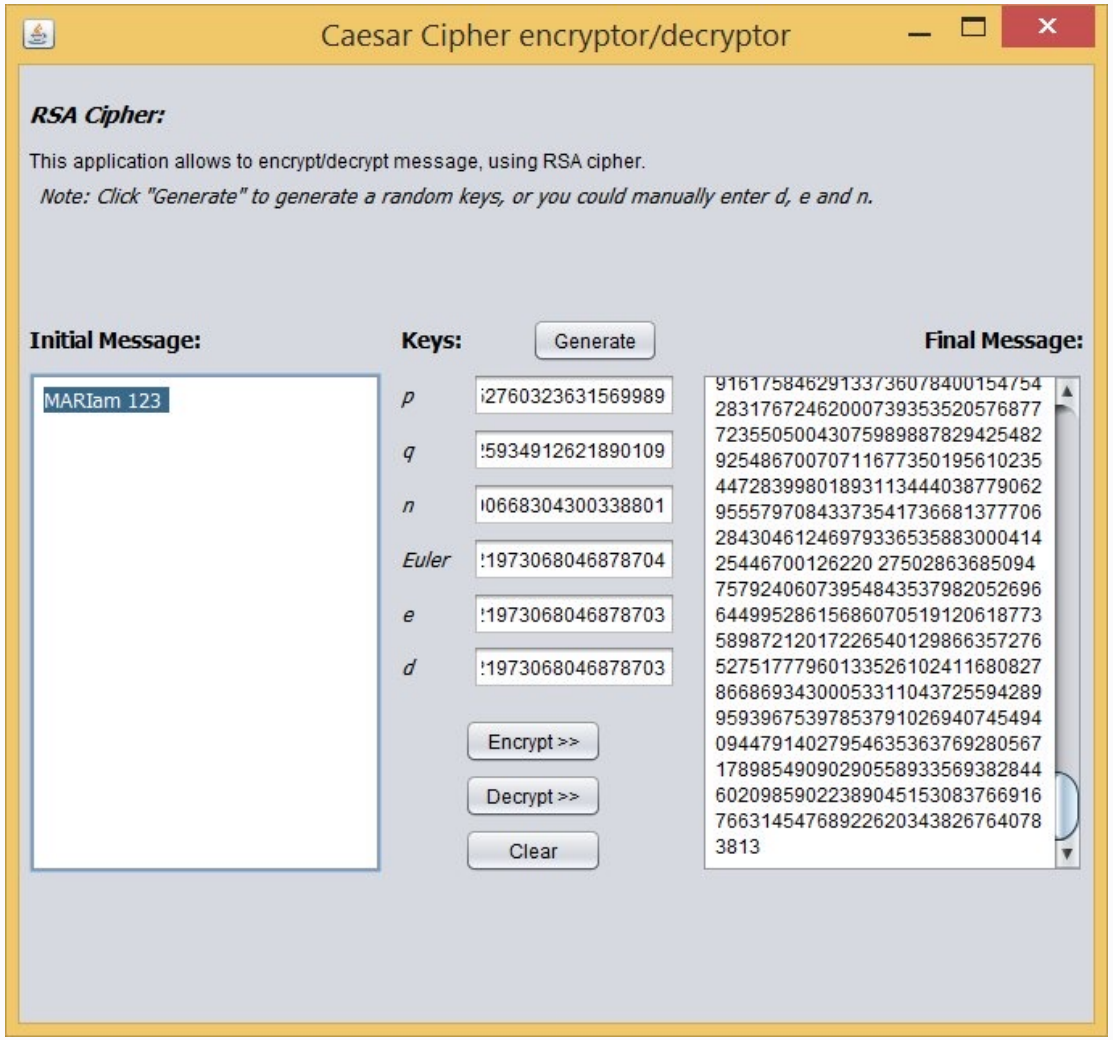
1. Caesar Cipher

Java, GUI application

A Caesar cipher is a type of substitution cipher in which each character in the plaintext is replaced by a character located some constant number of positions to the left or right of it in the alphabet. For example, in a shift 3 cipher, A would be replaced by D, B would become D, and so on.



Project #3



2. RSA Cipher

Java, GUI application

The application generates p and q numbers, the rest is automatically calculated according to the algorithm. The most important thing when we want to do encrypt in plaintext is that you can enter both text and numbers, but in ciphertext it displays only numbers when we press encrypt.

Project #3

More details about RSA Cipher

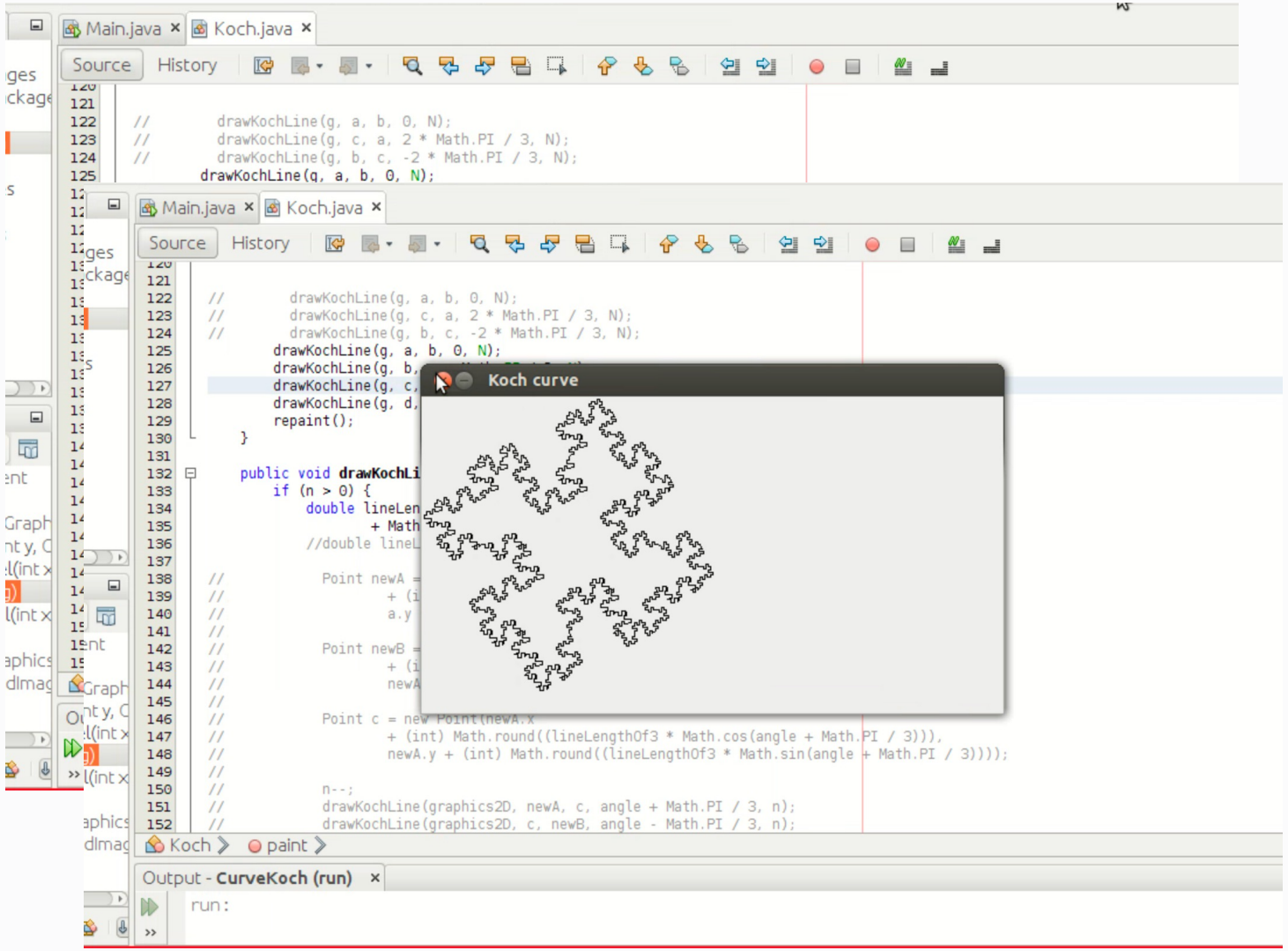
Firstly you need to generate public and private keys:

- Take two large primes p and q.
- Let's define n as the result of multiplying p on q (n= p*q).
- Let's choose a random number, which we will call d. This number must be coprime (have no common divisor other than 1) with the result of multiplication (p-1)*(q-1).
- Let's define a number e for which the following relation (e*d) mod ((p-1)*(q-1))=1 is true.
- Let's call the public key the numbers e and n, and the secret - d and n.

In order to encrypt data using the public key {e,n}, you need the following:

- Split the ciphertext into blocks, each of which can be represented as a number M(i)=0,1,2..., n-1 (i.e. only up to n-1).
- Encrypt the text, considered as a sequence of numbers M(i) according to the formula C(i)=(M(I)^e)mod n.

Project #4

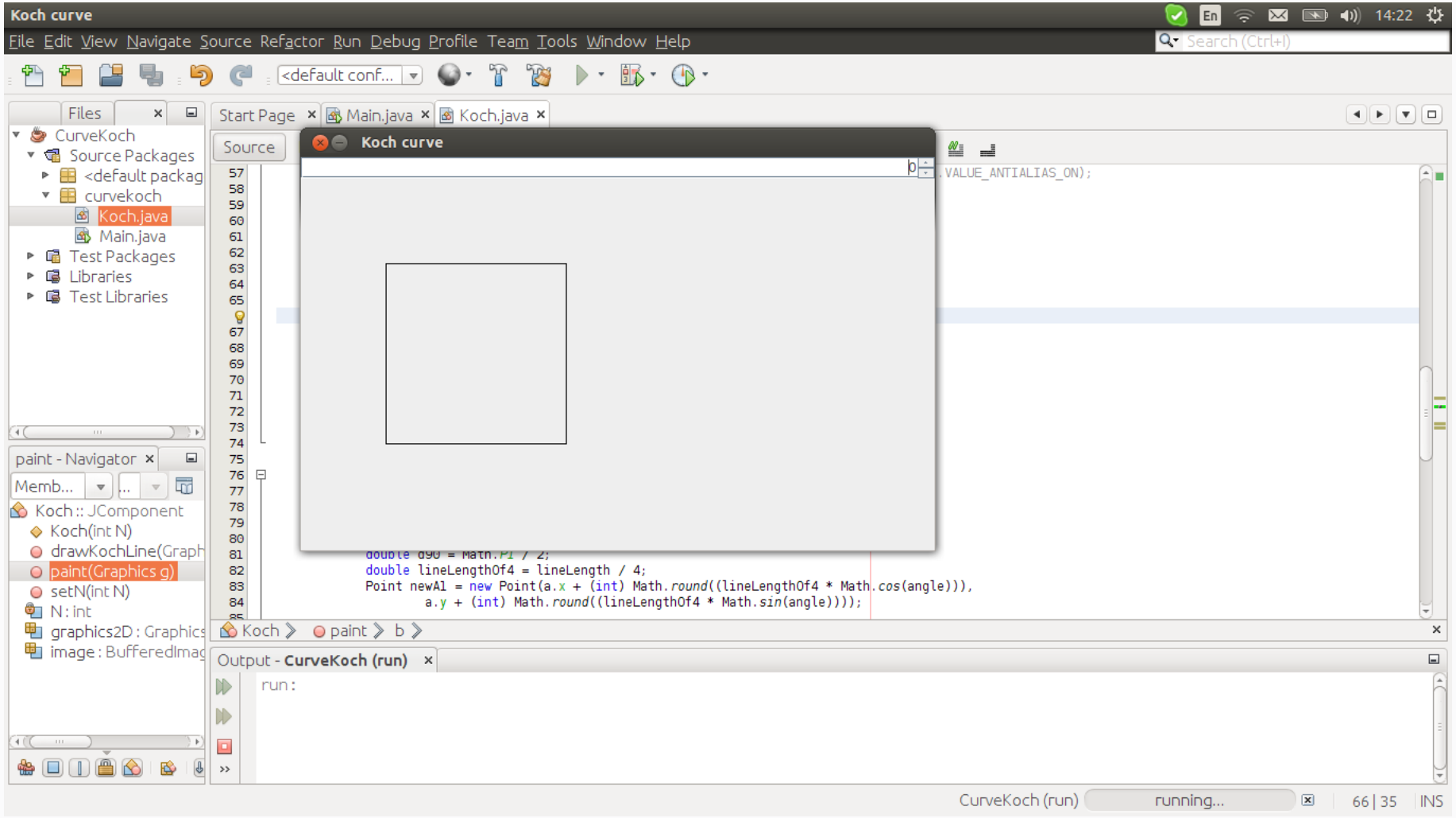


2. Curve Koch

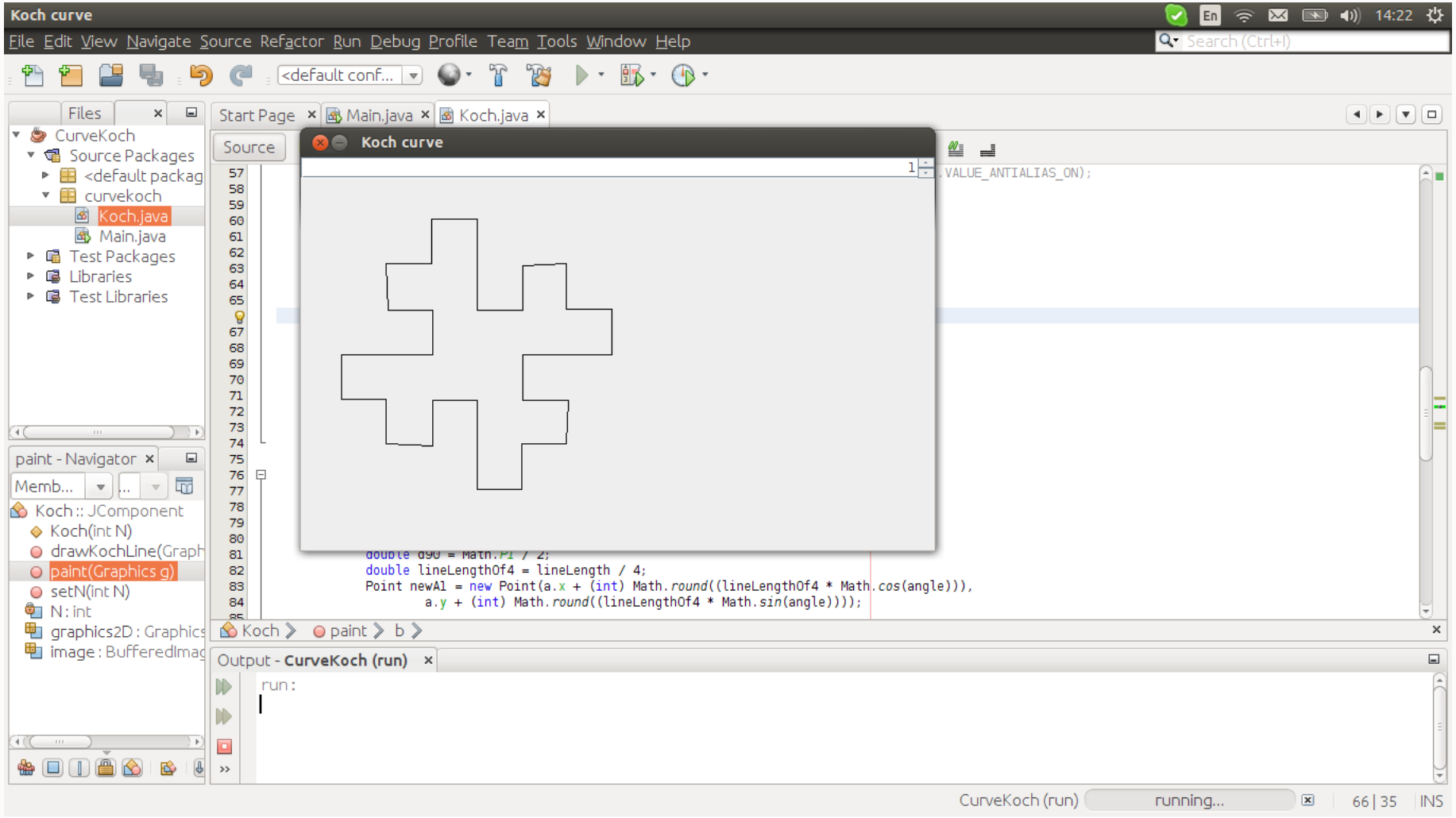
Java, GUI application

1. 0 – square
2. 1 – generator
3. 2 – Koch island
4. 3 – Koch island with colored borders

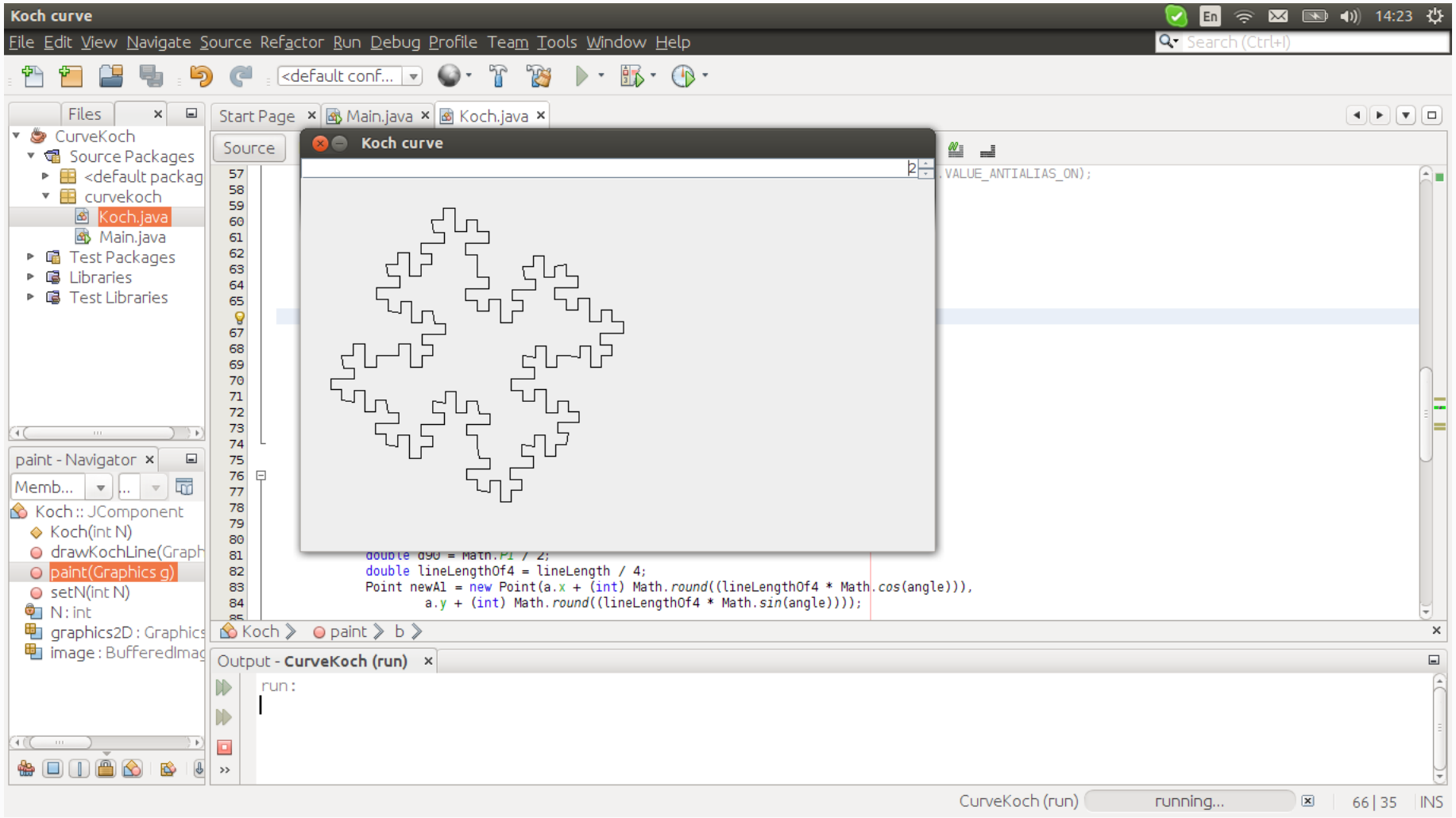
Project #4


Koch curve
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Files
CurveKoch
Source Packages
curvekoch
Koch.java
Main.java
Test Packages
Libraries
Test Libraries
Start Page
Main.java
Koch.java
Source
Koch curve
paint - Navigator
Koch :: JComponent
Koch(int N)
drawKochLine(Graph
paint(Graphics g)
setN(int N)
N : int
graphics2D : Graphics
image : BufferedImag
double d90 = Math.PI / 2;
double lineLengthOf4 = lineLength / 4;
Point newA1 = new Point(a.x + (int) Math.round((lineLengthOf4 * Math.cos(angle))),
a.y + (int) Math.round((lineLengthOf4 * Math.sin(angle))));
Output - CurveKoch (run)
run :
CurveKoch (run)
running...
66 | 35
INS

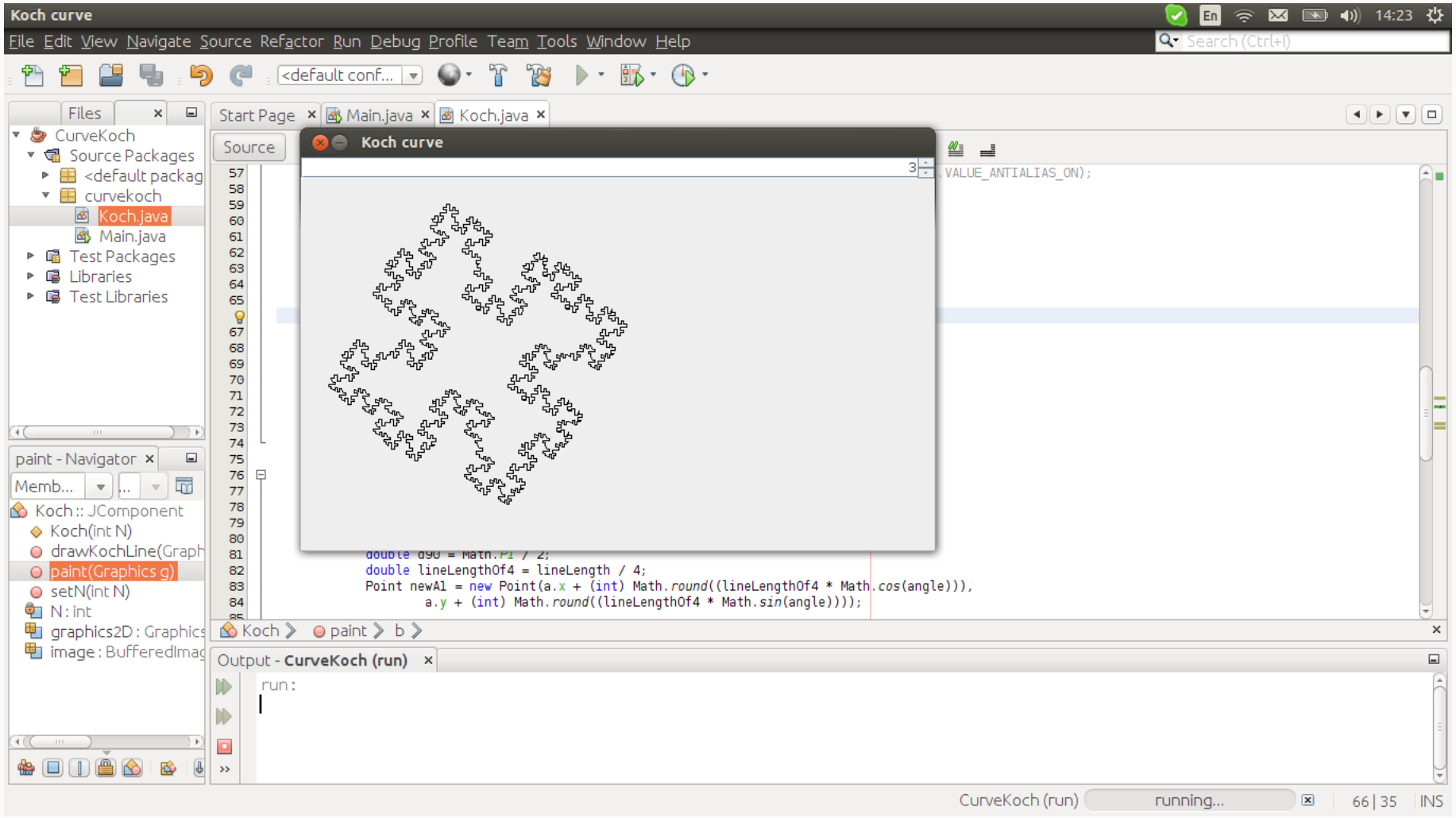
Project #4


Koch curve
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Files
CurveKoch
Source Packages
curvekoch
Koch.java
Main.java
Test Packages
Libraries
Test Libraries
Start Page
Main.java
Koch.java
Source
Koch curve
paint - Navigator
Koch :: JComponent
Koch(int N)
drawKochLine(Graph
paint(Graphics g)
setN(int N)
N : int
graphics2D : Graphics
image : BufferedImag
double lineLengthOf4 = lineLength / 4;
Point newA1 = new Point(a.x + (int) Math.round((lineLengthOf4 * Math.cos(angle))),
a.y + (int) Math.round((lineLengthOf4 * Math.sin(angle))));
Output - CurveKoch (run)
run :
CurveKoch (run)
running...
66 | 35
INS

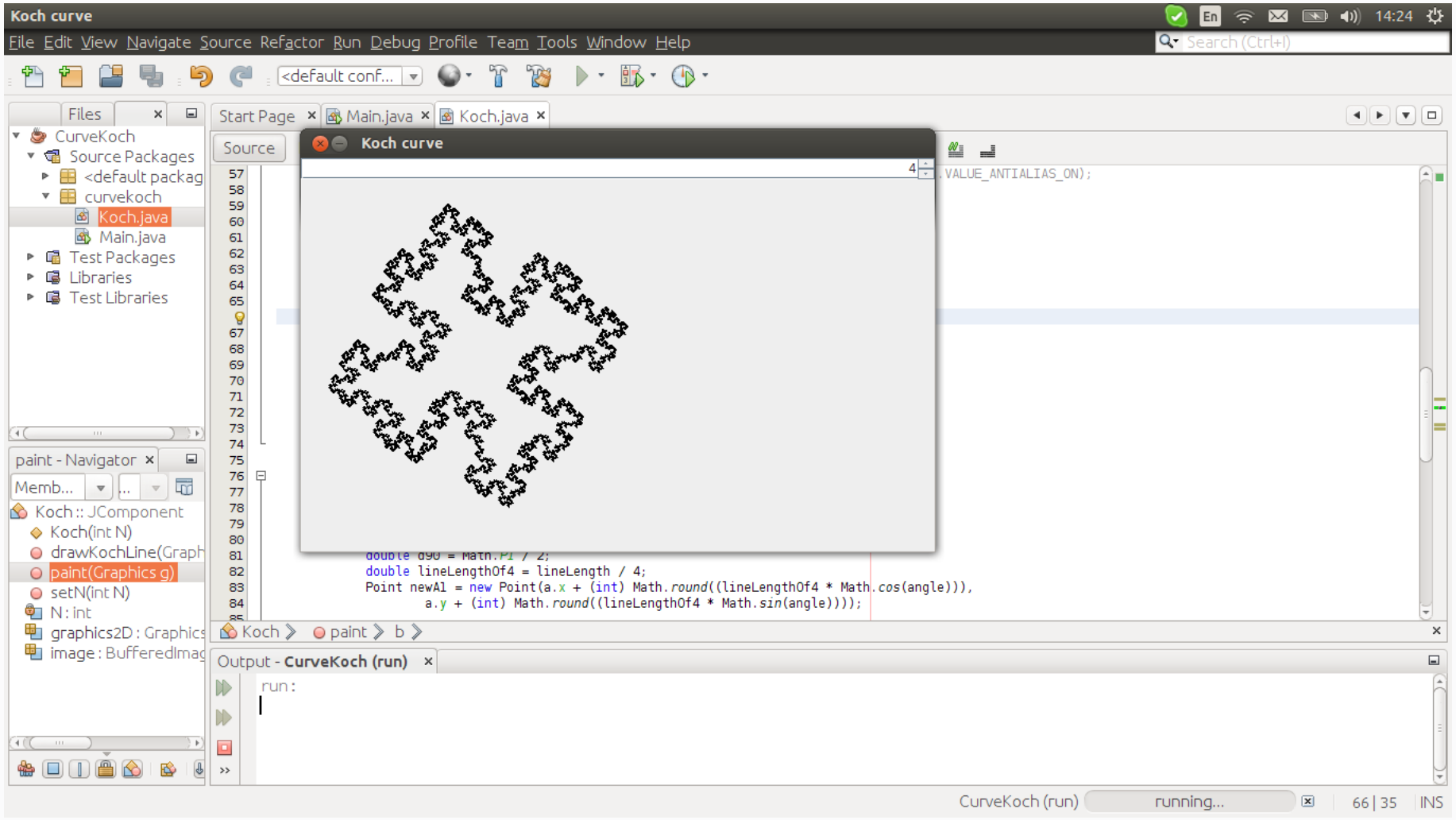
Project #4


Koch curve
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Files
CurveKoch
Source Packages
curvekoch
Koch.java
Main.java
Test Packages
Libraries
Test Libraries
Start Page
Main.java
Koch.java
Source
Koch curve
paint - Navigator
Koch :: JComponent
Koch(int N)
drawKochLine(Graph
paint(Graphics g)
setN(int N)
N : int
graphics2D : Graphics
image : BufferedImag
double d90 = Math.PI / 2;
double lineLengthOf4 = lineLength / 4;
Point newA1 = new Point(a.x + (int) Math.round((lineLengthOf4 * Math.cos(angle))),
a.y + (int) Math.round((lineLengthOf4 * Math.sin(angle))));
Output - CurveKoch (run)
run :
CurveKoch (run)
running...
66 | 35
INS

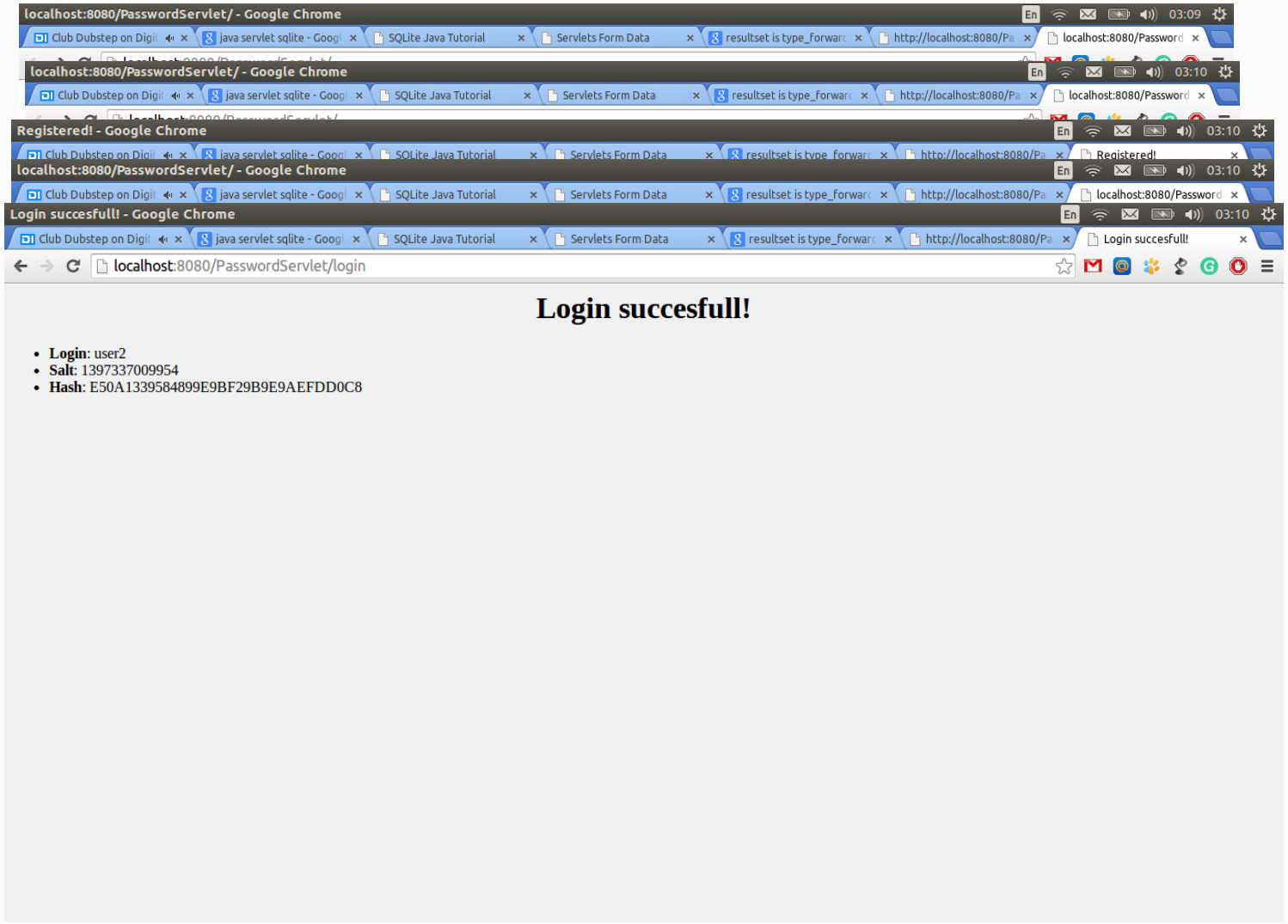
Project #4


Koch curve
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Search (Ctrl+I)
Files
CurveKoch
Source Packages
curvekoch
Koch.java
Main.java
Test Packages
Libraries
Test Libraries
Start Page
Main.java
Koch.java
Source
Koch curve
paint - Navigator
Koch :: JComponent
Koch(int N)
drawKochLine(Graph
paint(Graphics g)
setN(int N)
N : int
graphics2D : Graphics
image : BufferedImag
double d90 = Math.PI / 2;
double lineLengthOf4 = lineLength / 4;
Point newA1 = new Point(a.x + (int) Math.round((lineLengthOf4 * Math.cos(angle))),
a.y + (int) Math.round((lineLengthOf4 * Math.sin(angle))));
Output - CurveKoch (run)
run :
CurveKoch (run)
running...
66 | 35
INS

Project #4


Koch curve
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Files
CurveKoch
Source Packages
curvekoch
Koch.java
Main.java
Test Packages
Libraries
Test Libraries
Start Page
Main.java
Koch.java
Source
Koch curve
paint - Navigator
Koch :: JComponent
Koch(int N)
drawKochLine(Graph
paint(Graphics g)
setN(int N)
N : int
graphics2D : Graphics
image : BufferedImag
double lineLengthOf4 = lineLength / 4;
Point newA1 = new Point(a.x + (int) Math.round((lineLengthOf4 * Math.cos(angle))),
a.y + (int) Math.round((lineLengthOf4 * Math.sin(angle))));
Output - CurveKoch (run)
run :
CurveKoch (run)
running...
66 | 35
INS

Project #5

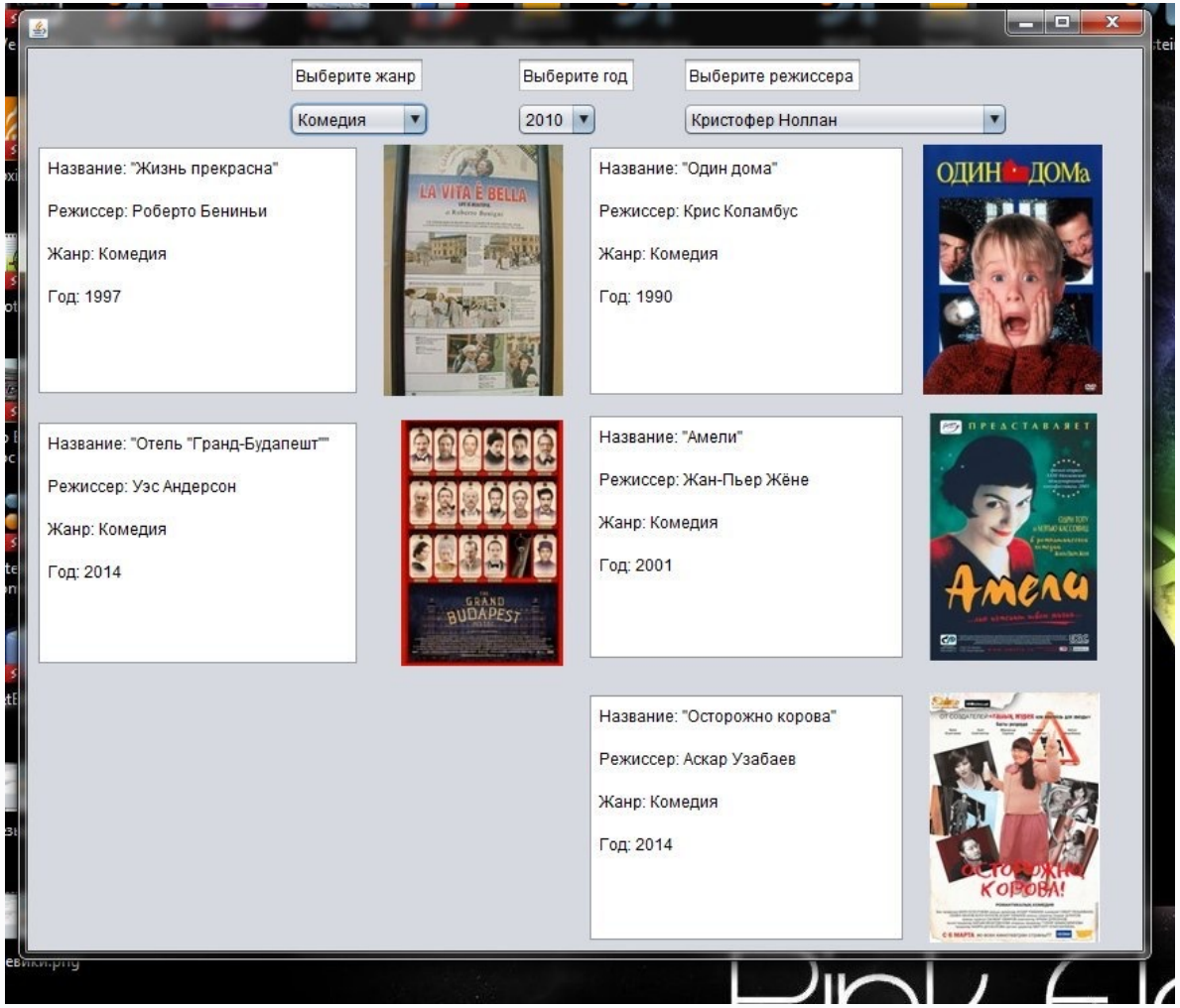


2. User Registration / Login

Java, GUI application

1. Generate an arbitrary password P Length 8 for example P=Pj6[iX_9
2. Take the machine time in miles seconds as salt for example S=1478007431
3. PS= P+S=Pj6[iX_91478007431
4. Use hash function get hash value (Message digest MD5) MD5(PS)->h
5. Save the salt and hash in the file
6. Verifying, enter id, enter the password generated by the given password method, and when we press the password, a message is displayed „Login successful“, or „User doesn‘t exist“, Password is incorrect

Project #6



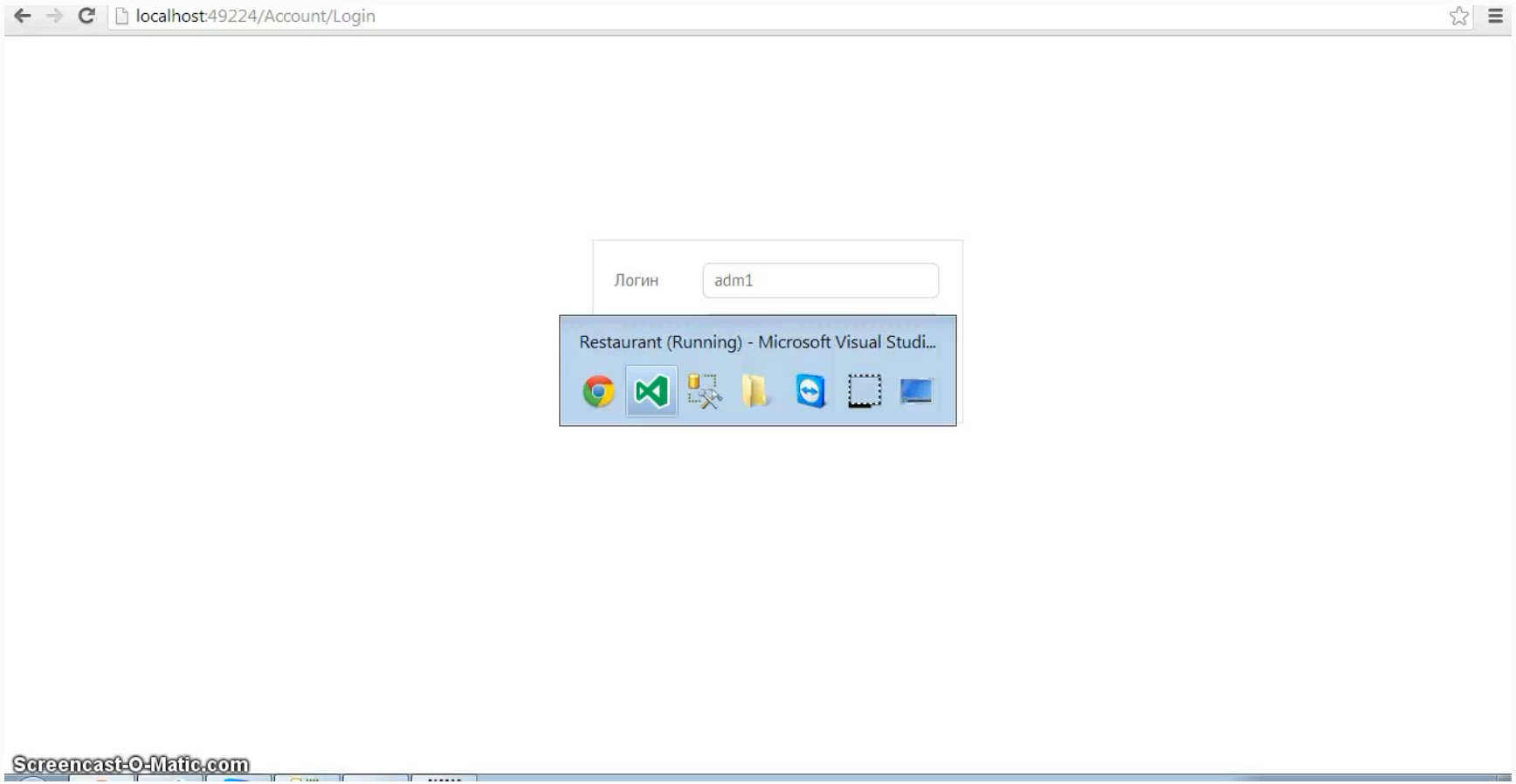
2. Movie Recommender System

Java, GUI application

GUI application filtering by genres, film directors and countries separately.
The user selects a certain genre, then GUI application displays a film of this genre only or selects a director displays films with this director, or selects which country displays films from this country, i.e., displays films on request

Project #7

Restaurant Web System



Project #7

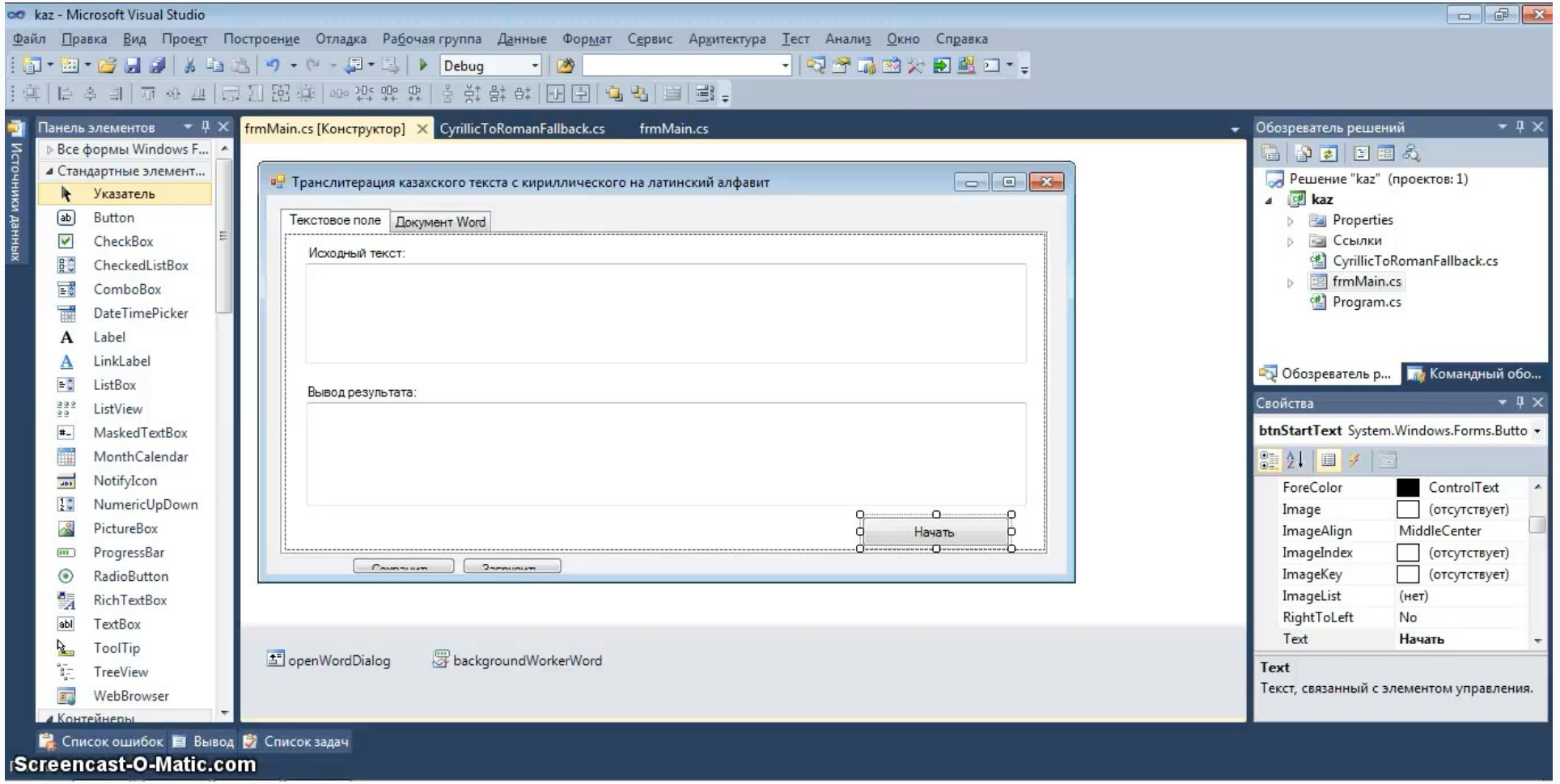
Restaurant Web System

HTML, PHP, SQL

Restaurant Web Application have Login panel. Activity tab has pages Tables and Waiters. Tables page has columns: Name (Table №1, Table №2 and etc.), Orders, Food preparing, Table decoration, Paid. Tables possible to see separetely. Inside of each table existing such columns as Name of dish, Quantity, Waiter, Time of order. Waiters page has columns: Name of waiters, his Orders, Preparation of food, Table Decoration, Paid.

Project #8

Kazakh language – Latin symbols transliteration system



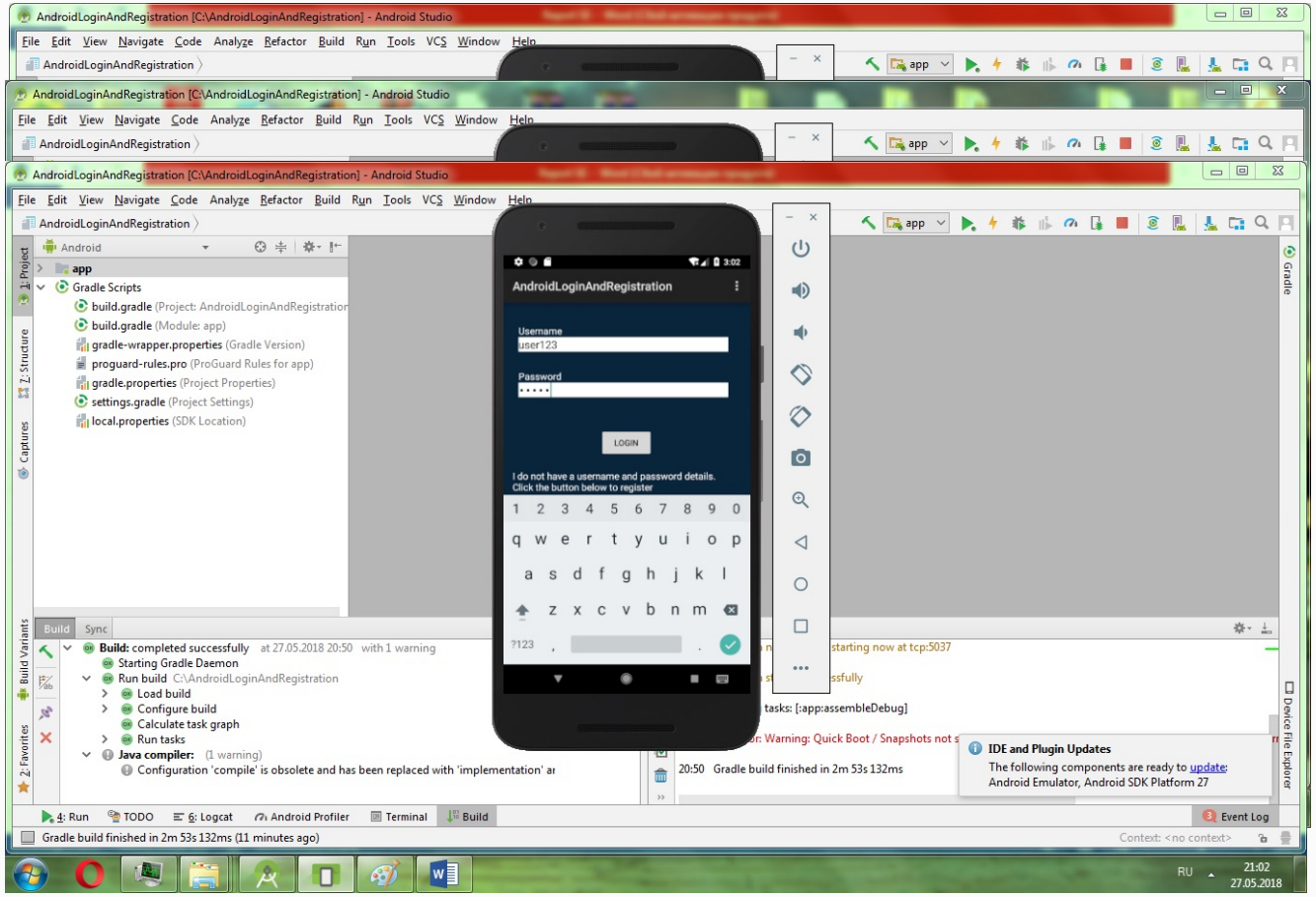
Project #8

Kazakh language – Latin symbols transliteration system

HTML, ASP.NET, C#

WEB application allows translate text in Kazakh language to Latin symbols. And this WEB application have opportunity to upload file with text in Kazakh language and translate, all text to Latin symbols.

Project #4



2. Android Login Registration

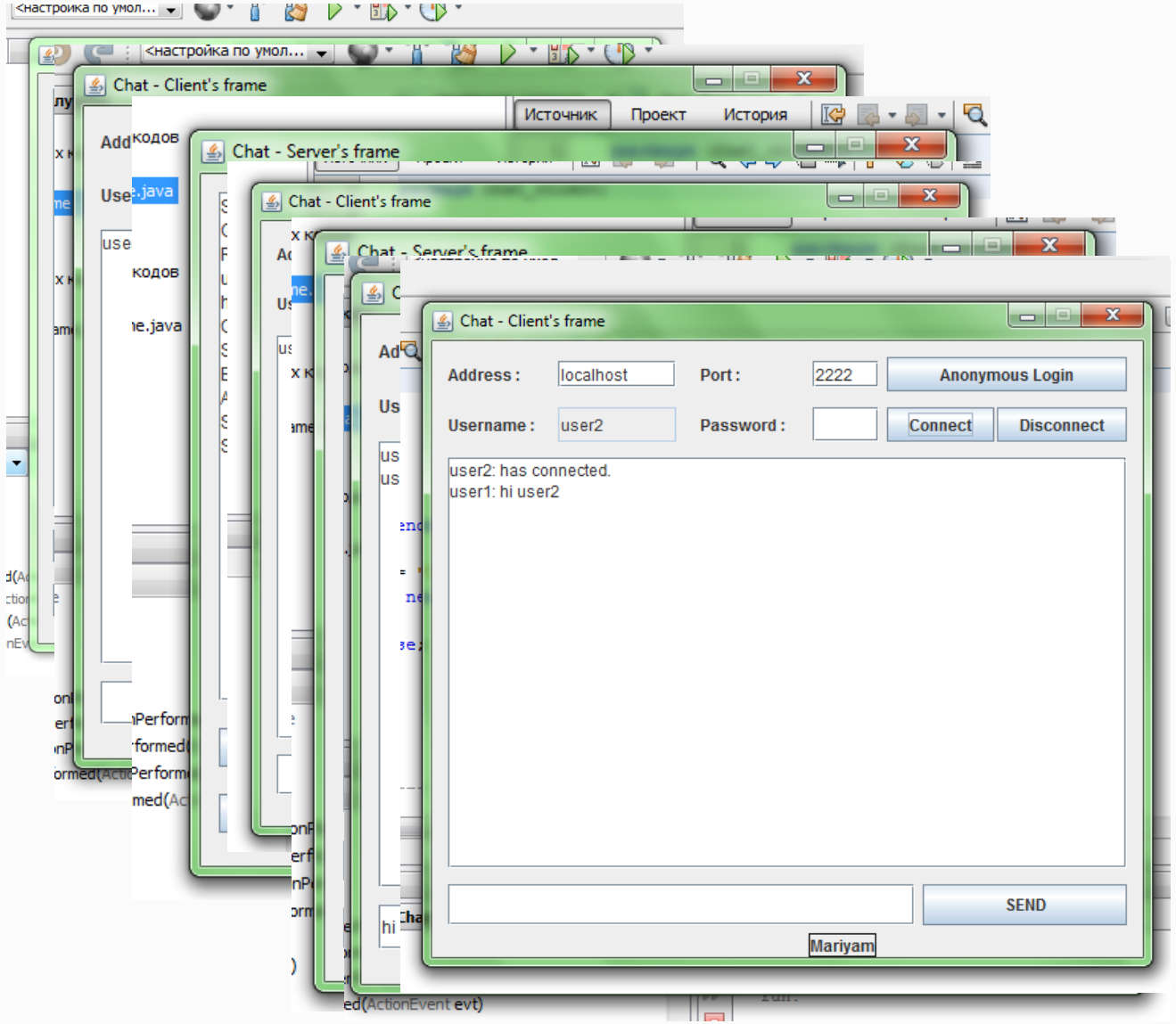
Android Studio, Apache Tomcat, Java PHP, MySQL

Java EE Web application

Steps of architect and design Java/Java EE web application:

- Design your database/entities first
- Use libraries sensibly and judiciously.
- Test application on all major browsers
- Select the application server based on its capabilities and more importantly 'your needs'. In this couse was used Apache Tomcat server.
- Use JPA interfaces
- Never abuse HttpSession.
- Keep entities Serializable. Enforce developers to use toString(), hashCode() and equals()

Project #4



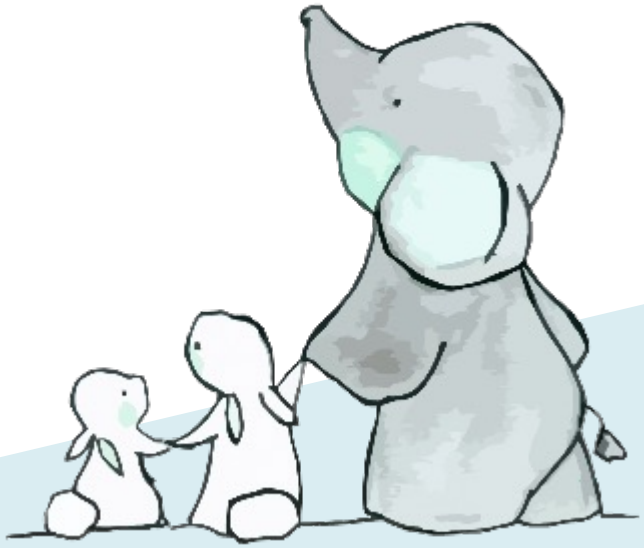
2. Clent – Server

Java GUI

1. Server is starting
2. Connection of user 1
3. Addition to server of user 1
4. Connection of user 2
5. Addition to server of user 2
6. Sending message of user 1 to user 2
7. Receiving message of user 1 to user 2

03 Software projects

during professional experience



研究成果

sanatel consulting

(1) Administration of CMS Moodle

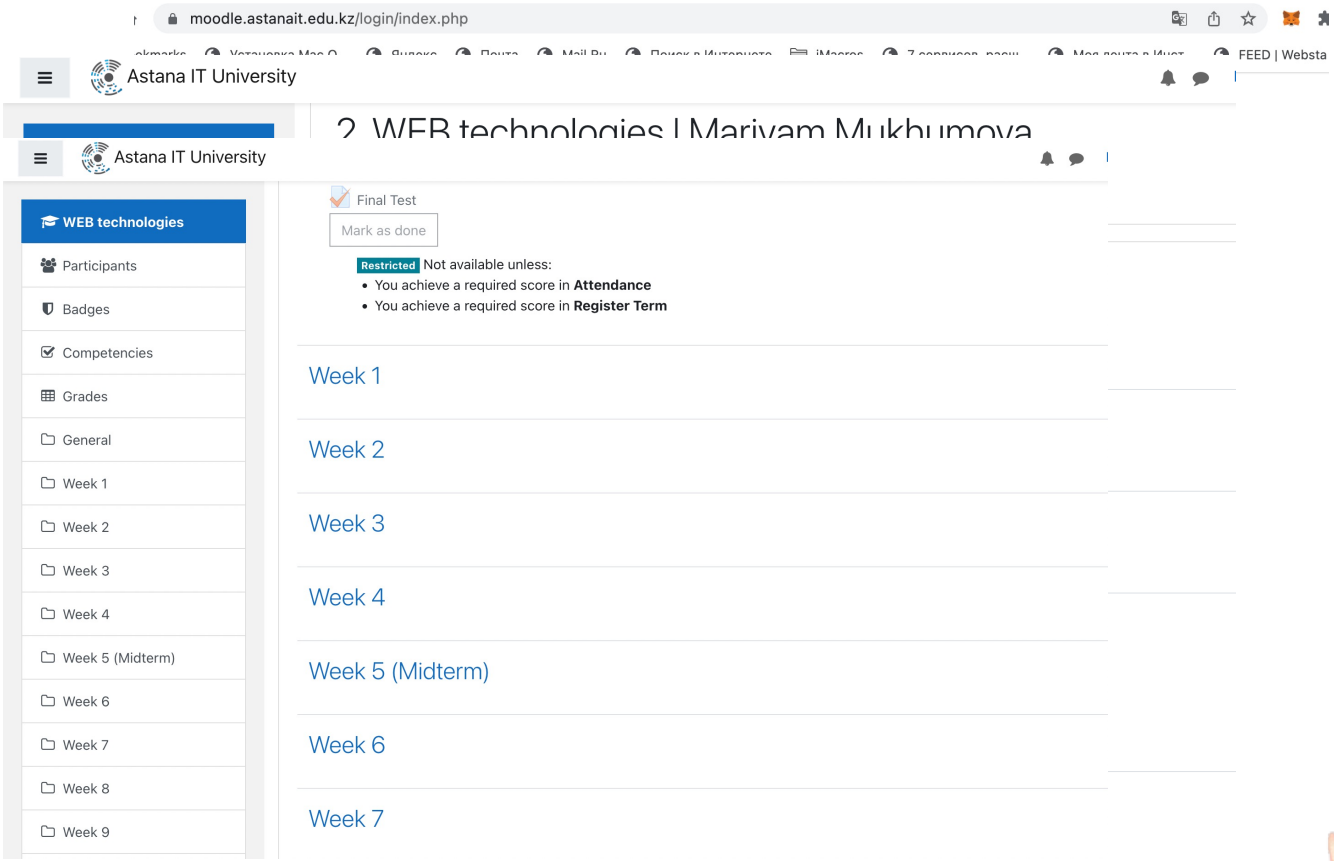
(2) Fixing Login/Logout

(3) Installation of Moodle

(4) Working with plugins

(5) Create courses

(6) Create tests



研究成果

(1) Administration of CMS DataLife Engine DLE

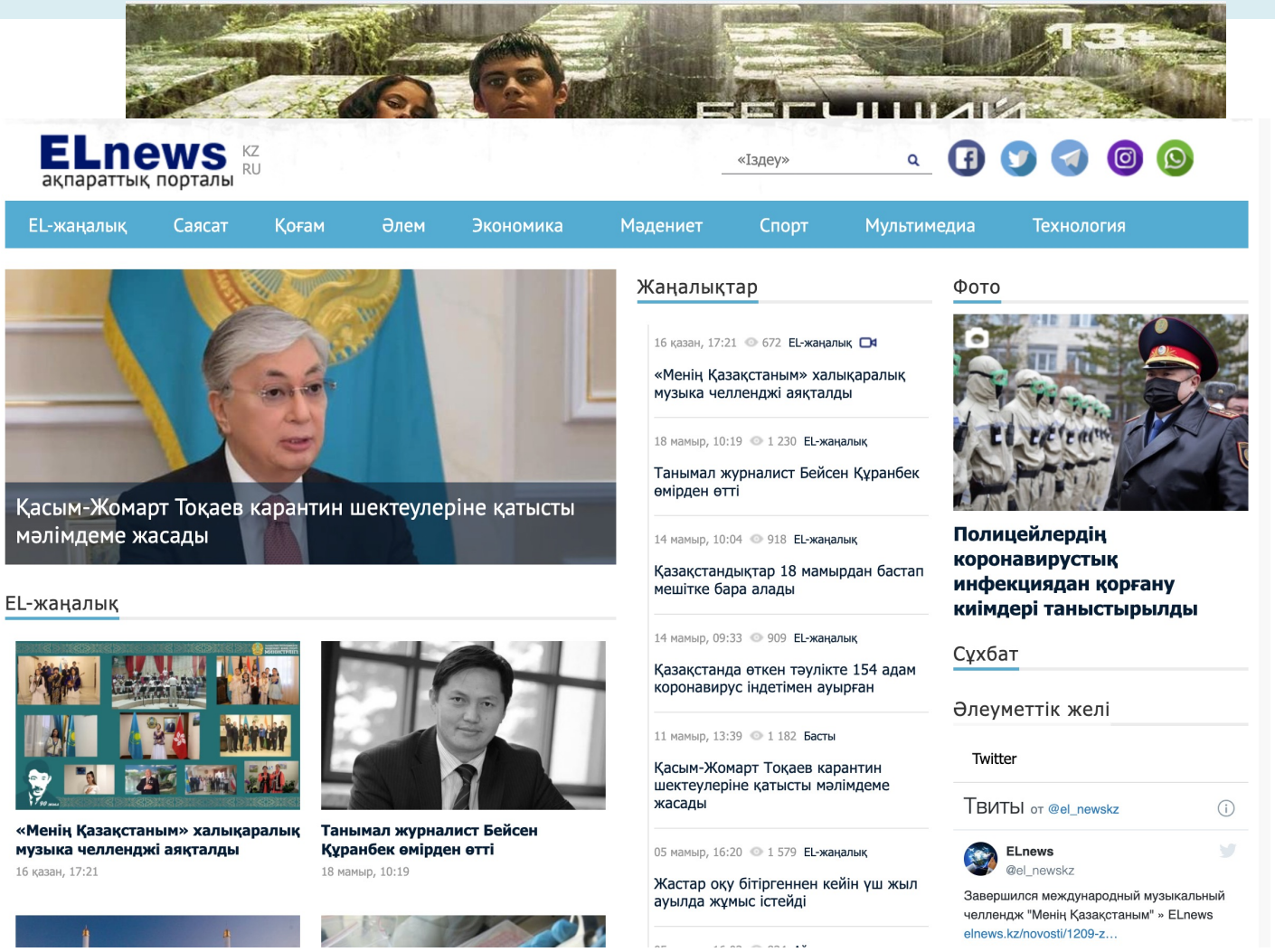
(2) Uploading content to website

(3) Changing backgrounds

(4) Writing posts

(5) Technical support of web-sites

(6) Creating of dynamic web-pages



研究成果

Beeline™

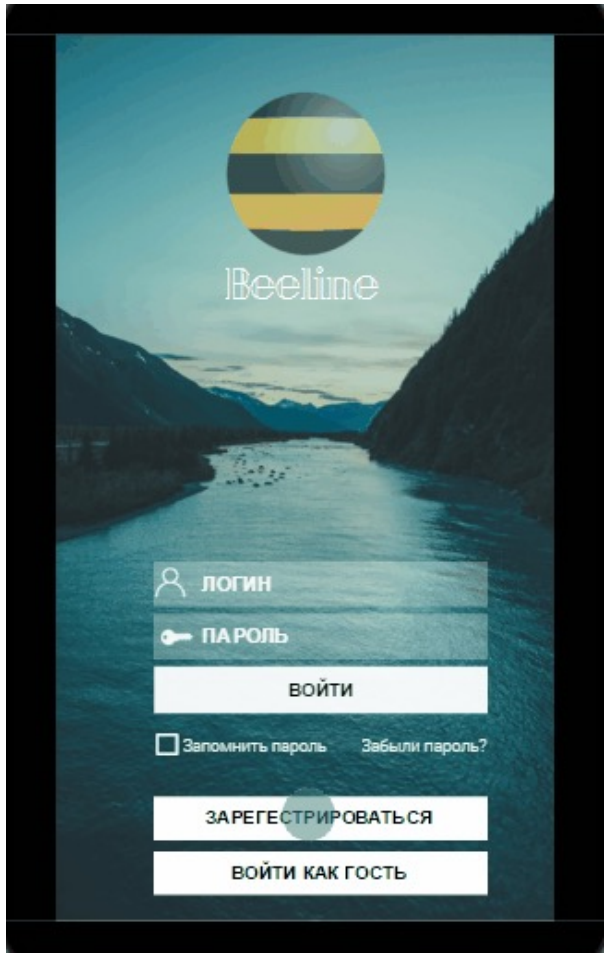
(1) UI/UX test

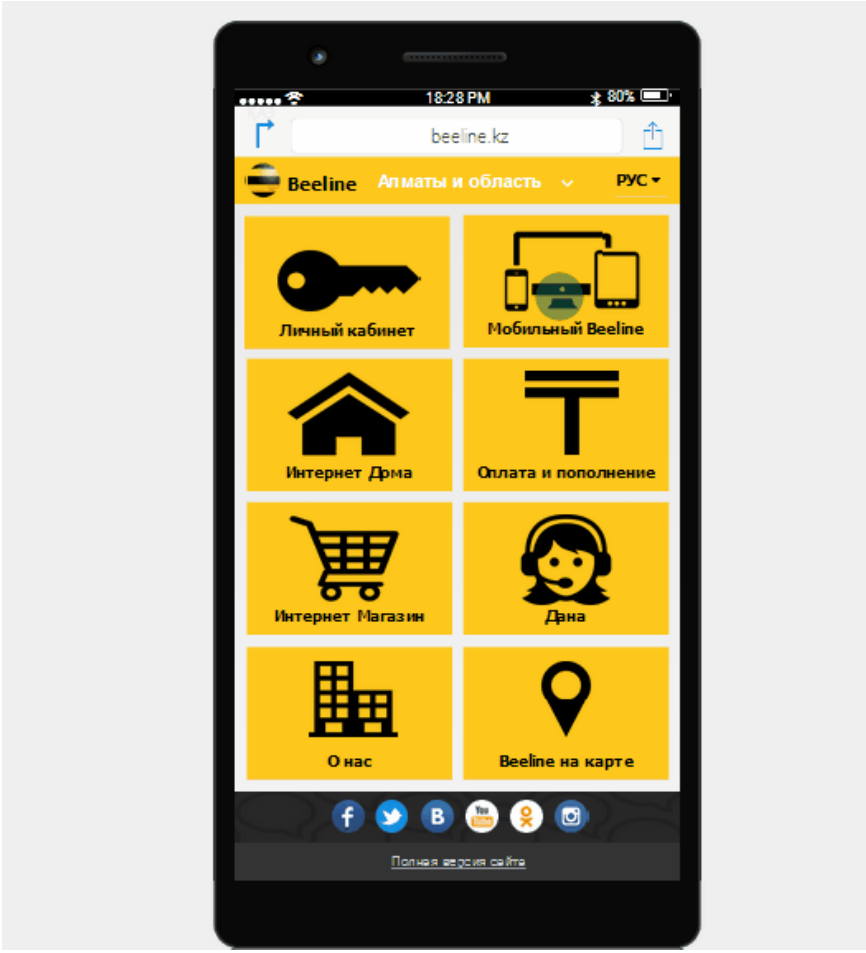
(2) Development of mobile app prototype

(3) Development of mobile website prototype

(4) Work with JustinMind

(5) Graphical design





研究成果

SAMSUNG MEDISON

(1) Design of uzi.kz website

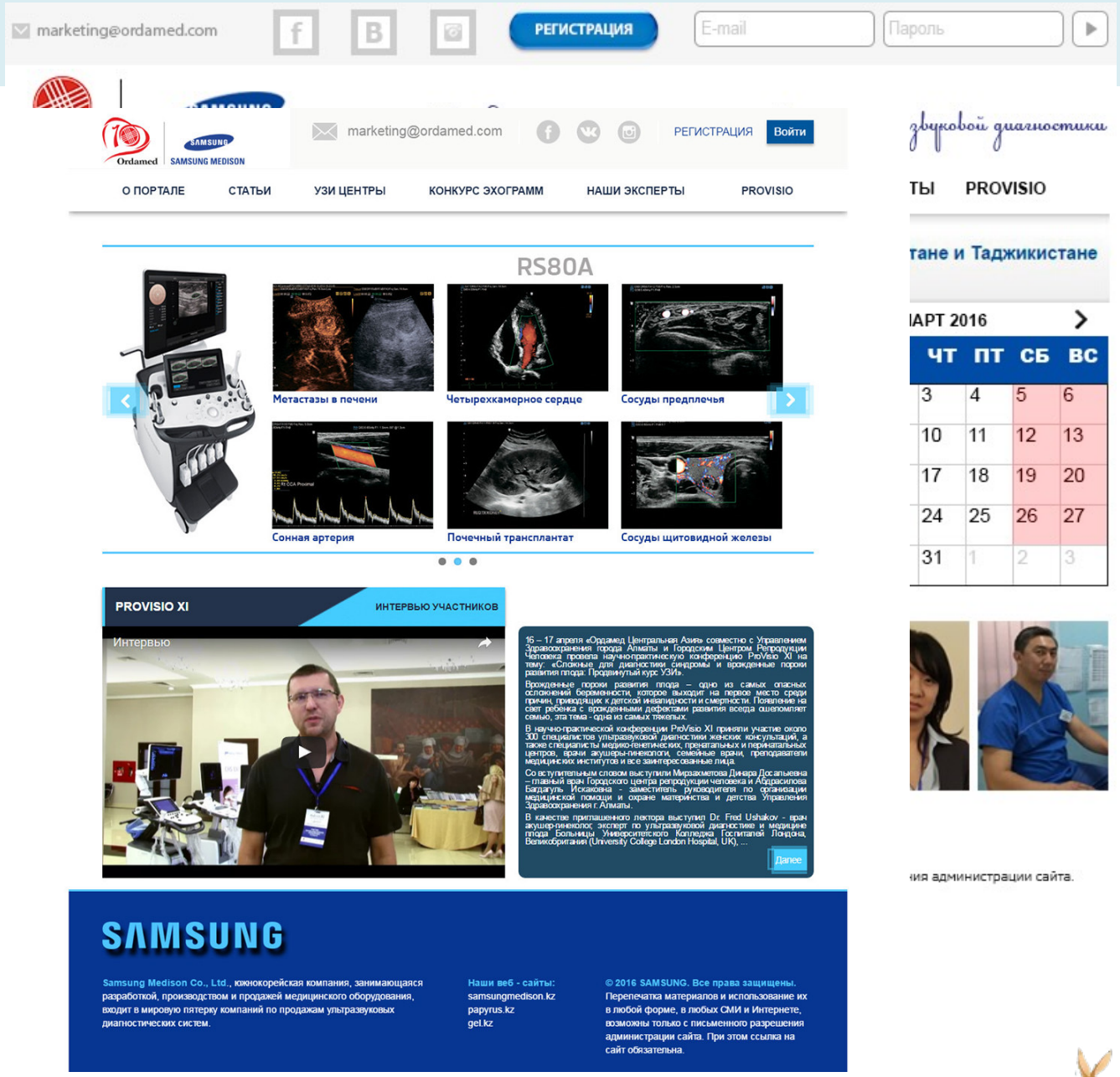
(2) Design of samsungmedison.kz website

(3) Graphical design

(4) Writing articles

(5) Uploading of content

(6) Installation of plugins



研究成果

Yu Ding Yu Tech.

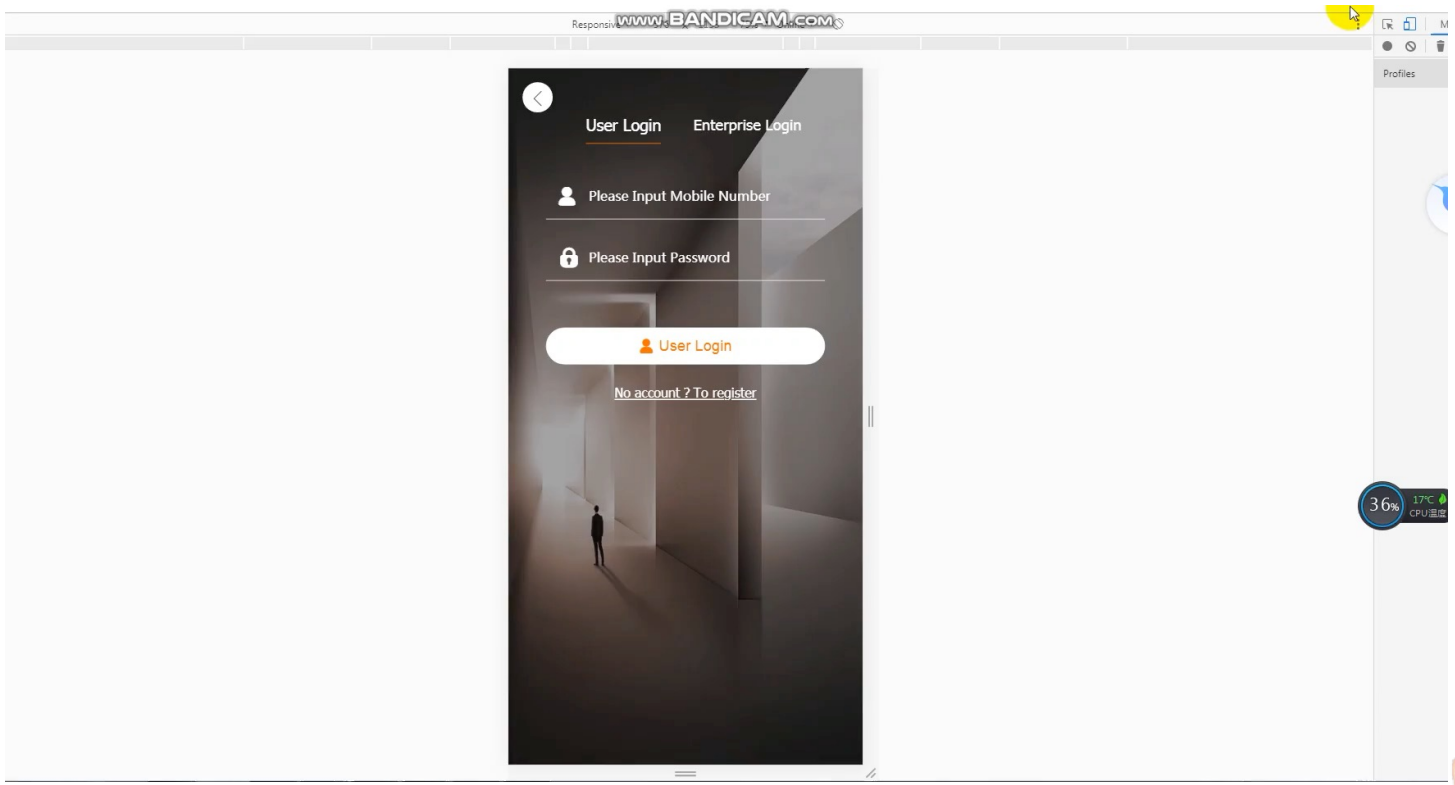
(1) Administration of online shops

(2) Create video instructions

(3) Create content for websites

(4) Writing documentation manual for users of websites

(5) Manage and tech.support of websites



研究成果

ASTANA IT UNIVERSITY

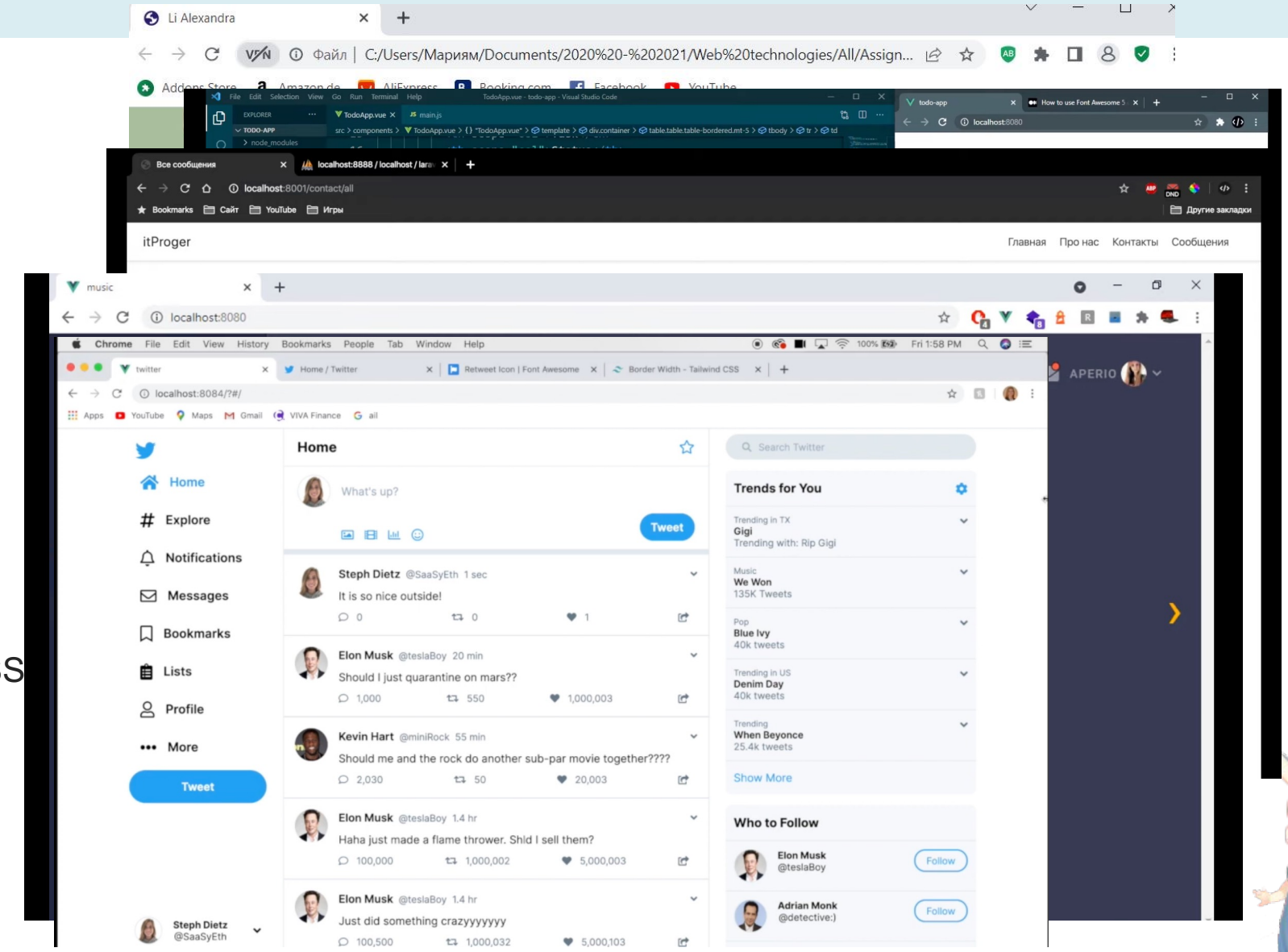
(1) Teach web-technologies

(2) Checking assignments

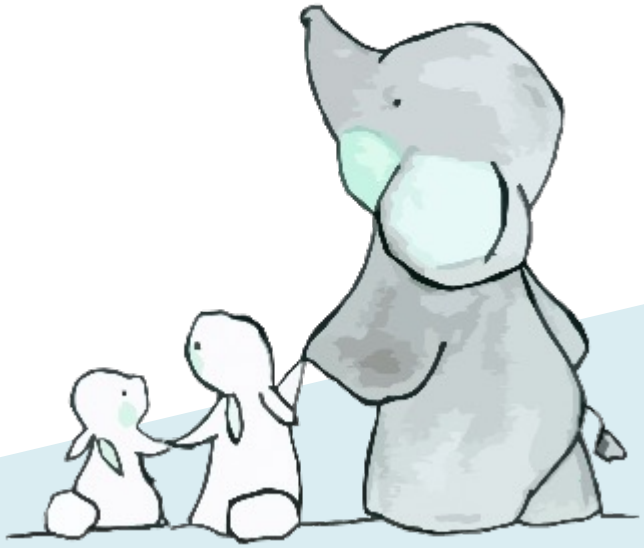
(3) Upload lectures/tasks on Moodle system

(4) Give materials about HTML, CSS JS, PHP, Vue.js, Laravel

(5) Record grades



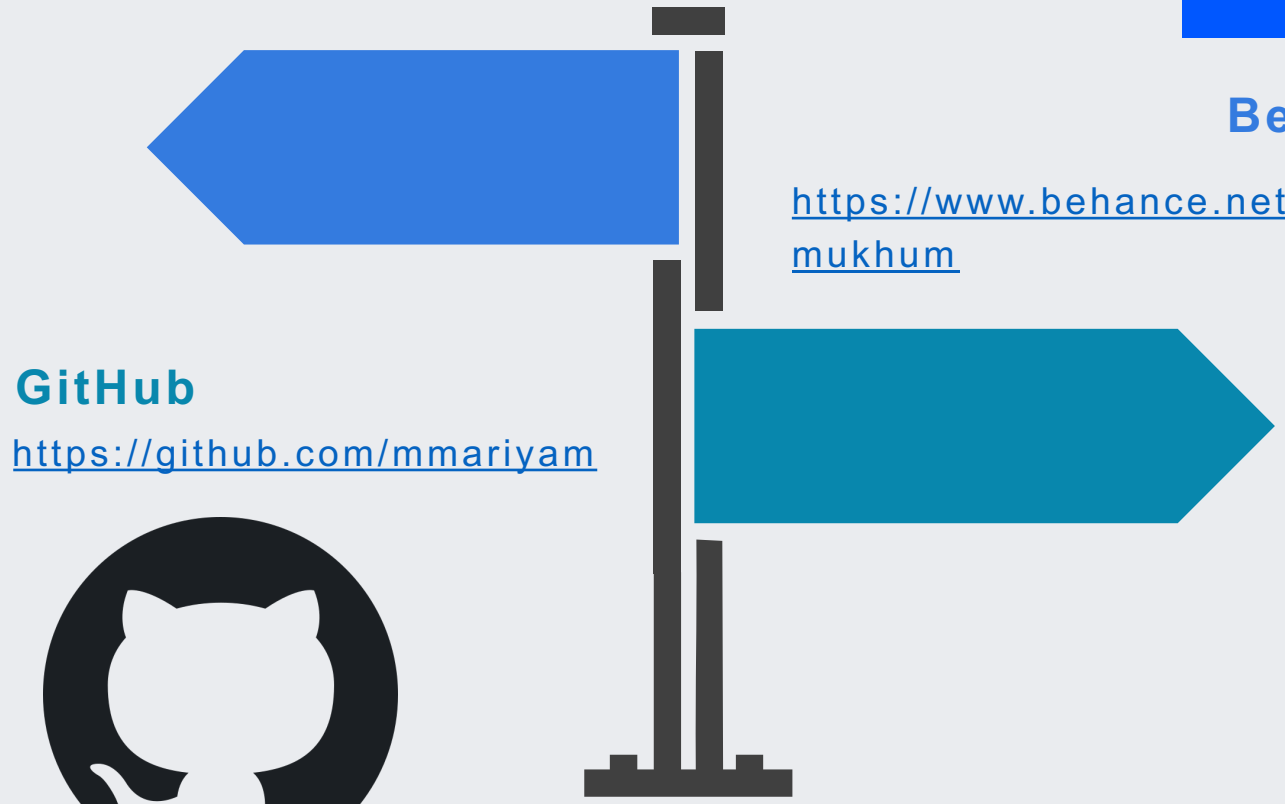
04 References



References

Behance

https://www.behance.net/mariyammukhum



GitHub

https://github.com/mmariyam

Thanks!