CS319 Term Project

Section 1

Group 1G

ReadyCodeGo

Project Analysis Report

Group Members:

- Mehmet Kaan Örnek (21901637)

- Göksu Şenerkek (21803341)

- Cem Apaydın (21802270)

- Ercan Bahri Nazlıoğlu (21903151)

- Mehmet Ali Öztürk (21703425)

Contents

1. Introduction

The application process for the Erasmus Exchange Program might be confusing and overwhelming for the coordinators due to the great number of applicants and their files and for the students due to physical submission to the different authorities. As the Group 1G, we wanted to ease the Erasmus Exchange Program application process for both students and the coordinators. The website's main aim is to ease and digitalize the application process by providing an environment where coordinators and students can continue their application process without getting lost within the files and emails.

This website will decrease the coordinators' work load by providing self regulating placement, easy access to the students' contact details and their files and an environment to handle all the processes. In addition, it will ease the students' application process by providing a medium that they can reach and submit their files to the different authorities easily and submit their desired courses. This system will also reduce the use of paper by replacing physical submission with digital submission.

On the other hand, this website aims to include the instructors and the faculty administration board members as well. Considering the course selection process, we aim to integrate instructors that are responsible for the similar courses, therefore, the interaction between the instructors and the applicants will be easier. Moreover, considering the pre-approval, post-approval and course transfer processes, integrating the faculty administration board members will ease the observation of the files and the processes of approving these.

The distinct features of the website will be the following:
- Self regulating placement of the applicants
- Automatic university offering to the applicants in the waiting list
- Access to any applicant's contact details
- File upload system
- Access to any applicant's documents
- Course selection and approval
- To do list for the coordinators

2.   Proposed System

2.1.   Functional Requirements

There will be four user types which are students, coordinators, instructors and faculty administration board members whose task is to give permission or sign specific application files.

2.1.1.   Login Page

When visitors enter the website, they will encounter the login page. Already registered users will be able to login with their email address and password that were specified during the registration.

If the visitors aren't registered, they will be able to register to the website through a button for signing up. During the registration, they will be asked for their Bilkent ID number, email address, password and their user type which are specified above. An extra keyword will be asked to check the stated user type which will be provided to the users that aren't students.

2.1.2.   Selected Applicants Page

The "Selected Applicants" page will be for the use of coordinators, instructors and faculty administration board members. On this page, the coordinators and faculty administration members will be able to see the applicants that are selected to the Erasmus Exchange Program in a table. Instructors will be able to see only the applicants whose course selection process needs the permission of that instructor as explained in "Course Selection " page.

Placement will be handled by this system without the need of the coordinators. Since the coordinators will be eliminated in this process, therefore, placement of the applicants will no longer be in their workload.

Each selected student will be placed in a row and they will be ordered by their total points. In each row, the student's name, surname, name of the university that student is placed in and the status of the approval will be provided. Beside the student's information, there will be a button for accessing the files. Using these, coordinators and faculty administration board members can access the documents like pre-approval form, and the instructors will be able to access the selected courses and syllabi of those courses. There will be a button for accessing the contact details of a student which is mainly the email address that was specified during the registration. In addition, there will be buttons for canceling the student's application for the coordinators' use only.

In case of a cancellation of any application, the quota of the university whose applicant withdrew will be updated, and the first student on the waitlist will be informed that there is an available university to apply. Therefore, coordinators will be able to keep track of the withdrawals and new selected students. The student whose application is canceled will see their application is canceled in the "Application Information" page.

2.1.3. Unselected Applicants Page

The "Unselected Applicants" page will be for the coordinators' use only as well. The table of the unselected applicants will be shown on this page.

This table will have the students ordered by their total points as well. Each row will contain the applicant's name, surname, contact details and the universities that student selected during the application.

The following cases cause problems for the self regulating placement system:
- A selected student decides not to attend the program, however, he or she doesn't state it in the application information page
- A selected student states that he or she won't attend the program in the application page, however, the student with the highest point in the waiting list doesn't state whether he or she is willing to attend the program

In such cases, some students might lose their chance to attend the exchange program. To eliminate these cases, the coordinators will be able to offer the universities with empty slots without waiting for them to state their status by the buttons that will be located in the students' rows. These offerings will show up in the "Application Information" for the applicants.

2.1.4. University Quotas Page

On the "University Quotas" page, the coordinator will be able to see the list of the host universities. In each row, the name and surname of the university, the link of the university's web page and the information of the coordinator that is responsible for that university will be provided. The number of students that are selected and empty slots for each university will be shown.

In addition to the information part, there will be buttons for adding and deleting a contact detail to the university, accessing the contact detail and changing the coordinator that is responsible for that university and taking notes as a reminder for specific universities. Using these features, the coordinators will be able to keep track of the universities and add the contact details of the host universities' coordinators. Since each university has a unique coordinator, they can select and change the universities that they are responsible for as well.

2.1.5. Application Information Page

The selected students will be able to see the university that they are placed in. The official website of the universities will be provided to the students.

There will be buttons for canceling their application. If a student cancels, the quota of that university will be updated, and the first student on the waitlist will be informed that there is an available university to apply.

The unselected students will face a text stating they are on the waiting list in this page. However, if a coordinator decides to offer a university with an empty slot or any of the selected students withdraws from the exchange program, the student with the highest point in the unselected applicants table will have an offer on this page. If the student accepts the offer, that student will move to the selected applicants table, and the available quota of that university will decrease. Therefore, coordinators will be able to keep track of the withdrawals and new selected students in "University Quotas" and "Selected Applicants" pages.

2.1.6. Course Selection Page

On the "Course Selection" page, students will be able to state the desired courses available in the host university and see if the selected course has previously taken from the previous exchange students.

If the selected course hasn't been taken from previous exchange students before, they will add the courses on Bilkent University that they think are similar and the mail of the instructor responsible for that course, and they will attach the course web pages. If the syllabus isn't on the course web page, and it is available as a document, they will upload the syllabus as explained in "File Upload System".

Once they are finished with the selection, both the coordinators and the instructors responsible for those courses will be able to see the selections and the information about the desired course at the host university. The applicants that finished their selection will be seen on the "Selected Applicants" page for the instructors as well as the coordinators.

The coordinators and instructors will be able to accept or reject each course. The students will be able to check if the courses are accepted or rejected on this page. They will be able to change the both rejected and accepted courses if they want, and the newly selected courses will be taken into consideration from the coordinators and instructors again.

2.1.7. File Upload System

The students will be able to upload their pre-approval form, post-approval form and learning agreement file to the system. In addition, they will be able to upload the syllabi of the desired courses of the host university in case the syllabi are available as documents. Using this functionality, the authorities will have access to these files using the buttons in the table on the "Selected Applicants" page.

2.1.8. To Do List

The coordinators and the faculty administration board members will be able to add tasks and notes to the list as a reminder. This functionality aims to combine the coordinators' notes with this environment.

As it is explained on the "Course Selection" page, when a selected applicant finishes their course selections, the instructors will have a notification in the to do list for the similar must courses, and the coordinators will have a notification in the to do list for the elective courses.

2.1.9. Profile Settings

To prevent inconvenient cases that applicants with the authorities cannot communicate due to the contact details that are no longer available, the users will have a settings page. Using this functionality, the user will be able to change their email address that they were specified during the registration. In addition, they will be able to add additional contact details to be on the safe side.

2.2.   Non-Functional Requirements

2.2.1.   User-Friendly Interface

Since the use of the website is limited to the exchange application. There is no need for a fancy design. It will be simple and effective. There won't be unnecessary features. Since each submission of the students is essential for their application, the submissions of the students and the way the authorities access them should be easy and organized.

2.2.2.   Easy Use and Customizability

Since the essential portion of the workload belongs to coordinators in this process, the user that will spend time the most will be the coordinators. Therefore, the website must be easy to use and customizable for them. They must be able to combine different aspects of their workload on this website.

2.2.3.   Inclusivity

Each process requires more than the interaction only between the applicants and coordinators. Therefore, this website must be inclusive for all kinds of authorities that are responsible for any kind of task in the application process. Their use of this website must be considered and be eased.

2.3. System Models

2.3.1. Use Case Model

2.3.2. Dynamic Models

2.3.2.1.   Activity Diagrams
- Pre-Approval Form Activity Diagram:

Two activity diagrams explain the preapproval and the post-approval process. In the first activity diagram for the preapproval process, Initially, the student logs in to the system, then uploads the requested documents to the system and chooses the courses they want to take with the help of a screen. If the required credit limit is not met in these courses, they must return to the course selection screen. When the process is complete, the information is updated, and the selected courses are sent to be checked by the instructor. If the courses are not approved by the teacher, the student has to choose the course again. After the lessons are approved by the teacher, the student prepares a pre-approval form and sends it to the coordinator. If this form is evaluated and approved by the coordinator, it is sent to the dean's screen for the dean's signature. If there is no problem and the process is successful, the final pre-approval form is sent to the foreign university by the coordinator.

- Post-Approval Form Activity Diagram:

The operation of the post-approval process in the second activity diagram is explained. At first, the transcript obtained from the host university reaches the coordinator. Then, the minimum passing grades are determined by the coordinator, and the pre-approval form submitted by the student to the system and the courses he/she chooses are checked. If this information in the incoming transcript is not confirmed, a new evaluation is made by the coordinator. If the transcript is approved, the course transfer form is prepared and made available to faculty administration board members. At this stage, this form is evaluated and the student's transfer is completed. The courses from the taken transcript are matched with Bilkent's courses.

2.3.2.2.   Sequence Diagrams
- ## Application Sequence Diagram

- ## Task Sequence Diagram

2.3.2.3. State Diagrams

Initial State

Upload
Courses

invalid courses

View Courses

Check
Courses

send message

Instructor /
Coordinator

Courses Valid

Uploading documents

Upload
Documents

invalid documents

Checking Documents

Check
Documents

All valid

Documents
Valid

Ready to Go

Final State

State Diagram Explanation:

Application: This is the initial state of the all process. It just ties up the approved or unapproved applications.

Application Successful: Students who are placed in a university will be in state if there is no cancellation. In case of cancellation, the student will go directly to the waiting bin.

Application Unsuccessful: Students who are not placed in any university will be in this state and they will be in the waiting bin.

Waiting Bin: Students who wait for an available space for a university will be in this list. They can get a message from the coordinator when there is an empty space.

Coordinator: This is a state for the students in the waiting bin in case of any cancellation.

Upload Courses: Students in the state of "Application Successful" will upload a course list that they will want to take.

Check Courses: Approved students make a course list for the coordinator and instructor to check if it is valid. If it is not valid, they go back to the upload state and reupload again.

Courses Valid: The state shows the course list is acceptable and the next state is for uploading documents.

Upload Documents: Documents will be uploaded for the student who has a valid course list.

Check Documents: The documents of the approved students, who have a course list will be checked in this stage.

Documents Valid: The state shows documents are acceptable and the next state is Ready to Go.

Ready to Go: This state is the final state and prerequisites are satisfied and students will be ready to go.

2.3.3. Object and Class Models

Register

+setName(string) : void
+setUserType(string) : void
+setPassword(string) : void
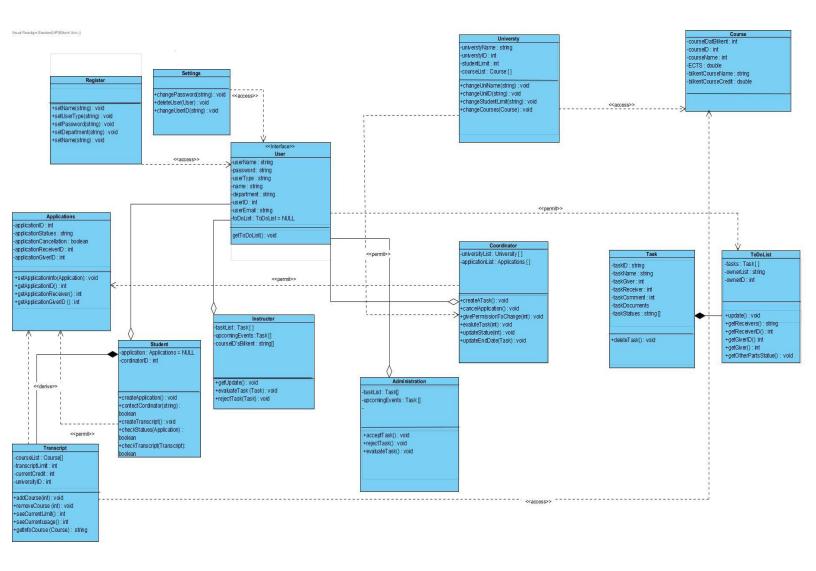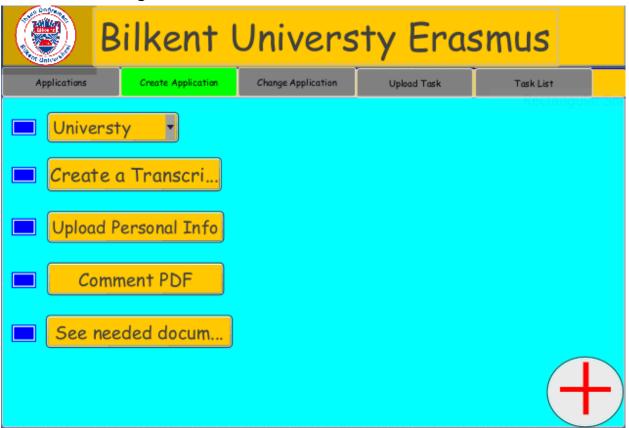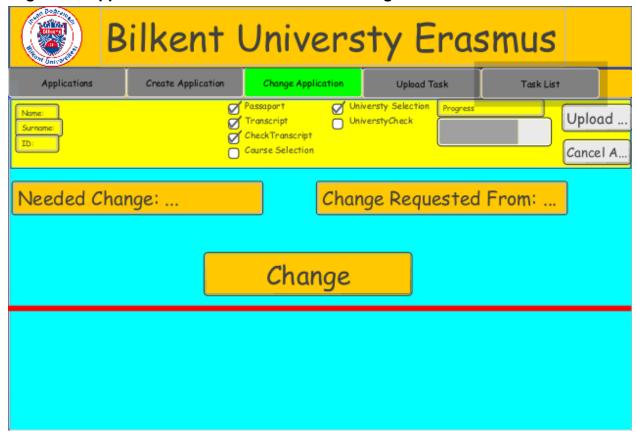+setDepartment(string) : void
+setName(string) : void

Settings

+changePassword(string) : void
+deleteUser(User) : void
+changeUserID(string) : void

<<access>>

Universty

-universtyName : string
-universtyID : int
-studentLimit : int
-courseList : Course [ ]

+changeUniName(string) : void
+changeUniID(string) : void
+changeStudentLimit(string) : void
+changeCourses(Course) : void

<<access>>

Course

-courseDatBilkent : int
-courseID : int
-courseName : int
-ECTS : double
-bilkentCourseName : string
-bilkentCourseCredit : double

<<Interface>>
User

-userName : string
-password : string
-userType : string
-name : string
-department : string
-userID : int
-userEmail : string
-toDoList : ToDoList = NULL

getToDoList() : void

<<access>>

<<permit>>

Applications

-applicationID : int
-applicationStatues : string
-applicationCancellation : boolean
-applicationReceiverID : int
-applicationGiverID : int

+setApplicationInfo(Application) : void
+getApplicationID() : int
+getApplicationReceiver() : int
+getApplicationGiverID () : int

<<permit>>

<<permit>>

Coordinator

-universtyList : Universty [ ]
-applicationList : Applications [ ]

+createATask() : void
+cancelApplication() : void
+givePermissionToChange(int) : void
+evaluteTask(int) : void
+updateStatue(int) : void
+updateEndDate(Task) : void

Task

-taskID : string
-taskName : string
-taskGiver : int
-taskReceiver : int
-taskComment : int
-taskDocuments
-taskStatues : string [ ]

+deleteTask() : void

ToDoList

-tasks : Task [ ]
-ownerList : string
-ownerID : int

+update() : void
+getReceivers() : string
+getReceiverID() : int
+getGiverID() :int
+getGiver() : int
+getOtherPartsStatue() : void

Instructor

-taskList : Task [ ]
-upcomingEvents :Task[]
-courseID'sBilkent : string[]

+getUpdate() : void
+evaluateTask (Task) : void
+rejectTask(Task) : void

Student

-application : Applications = NULL
-cordinatorID : int

+createApplication() : void
+contectCordinator(string) :
boolean
+createTranscript() : void
+checkStatues(Application) :
boolean
+checkTranscript(Transcript):
boolean

Administration

-taskList : Task[]
-upcomingEvents : Task []
-

+acceptTask() : void
+rejectTask() : void
+evaluateTask() : void

<<derive>>

<<permit>>

Transcript

-courseList : Course[]
-transcriptLimit : int
-currentCredit : int
-universityID : int

+addCourse(int) : void
+removeCourse (int) : void
+seeCurrentLimit() : int
+seeCurrentusage() : int
+getInfoCourse (Course) : string

<<access>>

15

2.3.4.   User Interface - Screen Mock-ups

Student Profile Page

Page for reapplication of the student in the Waiting Bin

Bilkent Universty Erasmus
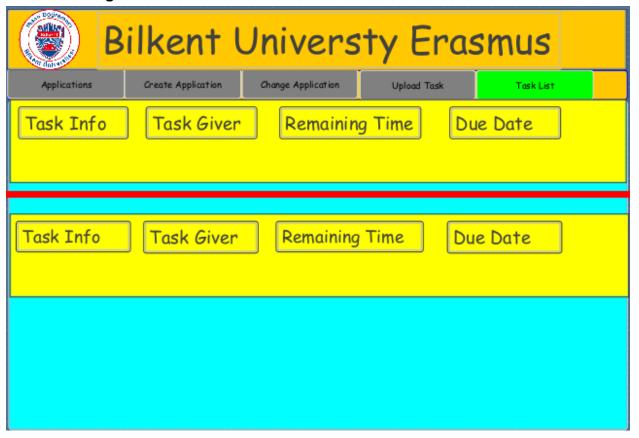
| Applications | Create Application | Change Application | Upload Task | Task List |

Name:
Surname:
ID:

☑ Passaport
☑ Transcript
☑ CheckTranscript
☐ Course Selection

☑ Universty Selection
☐ UniverstyCheck

Progress

Upload ...

Cancel A...

Needed Change: ...

Change Requested From: ...

Change

17

Sign Up Page

Login Page

Bilkent Universty Erasmus

ID :

Password :

LogIn

Register

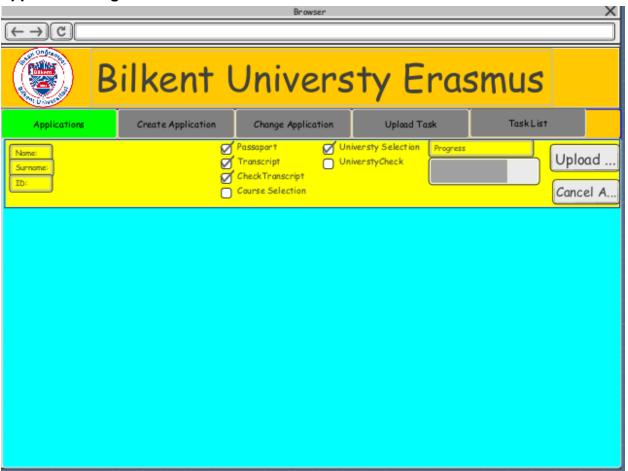**To Do List Page**

Bilkent Universty Erasmus

| Applications | Create Application | Change Application | Upload Task | Task List |

Task Info   Task Giver   Remaining Time   Due Date

Task Info   Task Giver   Remaining Time   Due Date
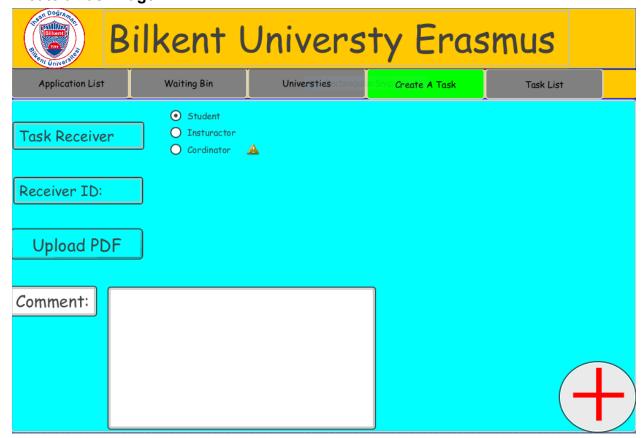
Uploading Document Page

Application Page

Selected Applicants Page

Bilkent Universty Erasmus

| Application List | Waiting Bin | Universties | Create A Task | Task List |

Name:
Surname:
ID:

☑ Passaport
☑ Transcript
☑ CheckTranscript
☐ Course Selection

☑ Universty Selection
☐ UniverstyCheck

Progress

Upload ...
Cancel A...

Name:
Surname:
ID:

☑ Passaport
☑ Transcript
☑ CheckTranscript
☐ Course Selection

☑ Universty Selection
☐ UniverstyCheck

Progress

Change
Cancel A...

Name:
Surname:
ID:

☑ Passaport
☑ Transcript
☑ CheckTranscript
☐ Course Selection

☑ Universty Selection
☐ UniverstyCheck

Progress

Change
Cancel A...

Unselected Applicants Page

Bilkent Universty Erasmus

| Application List | Waiting Bin | Universties | Create A Task | Task List |

Name:
Point:
SurName
Bilkent Universty
ID:
Email:

Give Permision to Chan...

Name:
Point:
SurName
Bilkent Universty
ID:
Email:

Give Permision to Chan...

Name:
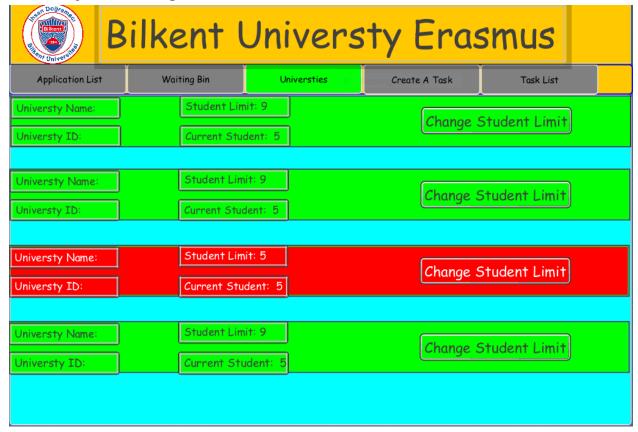Point:
SurName
Bilkent Universty
ID:
Email:

Give Permision to Chan...

Create a Task Page

University Quotas Page

3. Glossary & References

-http://cs.bilkent.edu.tr/~exchange/procedure.html

-Object-Oriented Software Engineering, Using UML, Patterns, and Java, 3rd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2010, ISBN-10: 0136066836.

-Alkan, Can (2022) " Applications for Erasmus and Exchange [PowerPoint slides] Computer Engineering, Bilkent University.