CS 319-1 Group 8

Deliverable 2 - 1st Iteration

Bilkent Tanıtım Ofisi Management System

Short-name: BILFO

27.10.2024

Ata Uzay Kuzey   22203050

Duru Solakoğlu   22102720

Emir Görgülü   22202834

Kerem Demirören   22203380

Nazlı Apaydın   22202104

1. Non-Functional Requirements

1.1 Usability

The project BILFO aims to enhance the current information office's (Bilkent Tanıtım Ofisi) process by offering a clear and intuitive system, particularly for Özhan Örge, coordinators, advisors, guides, and visitors. The system should ensure that BTO can complete its tasks effortlessly through a digital platform. To ensure the user interface is clear and straightforward:

- The system will have a very simple interface in order to make users adapt to it easily.
- To increase the look & feel of the system and to make the users feel like a part of Bilkent society, Bilkent colors will be used (e.g., Pantone reflex blue (RGB: 0, 20, 137), red of flag (RGB: 227, 10, 23), black (RGB: 0, 0, 0) and white (RGB: 255, 255, 255)) for a user-friendly experience.
- Immediate feedback will be provided to users of the system. For instance, the outside user will get a clear message if the application form has a piece of missing information.
- Guides and advisors will be able to arrange their schedules through their profiles.
- Communication will be fast and easy owing to the automated mail system.
- Tour assignments to guides will be done in two ways. An advisor will ask for guides or guides will volunteer. This will make the progress faster.

1.2 Security/Safety

Bilfo stores information about students, high schools, and councilors, all of which should be kept secure. To ensure this, we will use the following strategies.

- All of the information within Bilfo will be inaccessible to outside users.
- Information about schools and councilors will only be visible to the coordinator, acting director, and admins.
- Users will be signed up only by trusted individuals in the system, like advisors, coordinator, acting director, and admins.
- Instead of storing passwords directly, their salted hash values will be stored.
- For more powerful actions that are done by admins, a password is required for every action.
- Log-in attempts can only be done up to 5 times an hour.

1.3 Performance

The aim of this project is to make things easier and faster for arranging tours and evaluating fair invitations Bilkent Information Office handles.

- The emails to outside users will be sent automatically in order to lighten the Information Office's burden.
- Arranging tours and guides will be faster owing to schedule application.
- Visitors will experience more organized tours in the future, owing to evaluation forms that include feedback from past tours' counselors.

1.4 Robustness

In this project, we aim to create a user-friendly environment for the Bilkent Information Office, high schoolers and the counselors of the high schools who are interested in Bilkent. It is important to us that the users will use the site without encountering any bugs that can be caused by wrong inputs and receiving incorrect data.

- All of the inputs in the application forms will be checked by a type checker. For example, a visitor count cannot be a word, it can only be an integer.
- Some of the inputs in the application forms will be checked for their legitimacy. For example, the school name must be a school that actually exists.

1.5 Maintainability

Bilfo should be designed to be a maintainable software project. To ensure this, we will abide by these credentials.

- Bilfo's code will be written with easily understandable function and variable names.
- Bilfo will be created in a way that allows trusted users to change the information in the site without changing the code.
- Bilfo's code will feature extensive documentation.

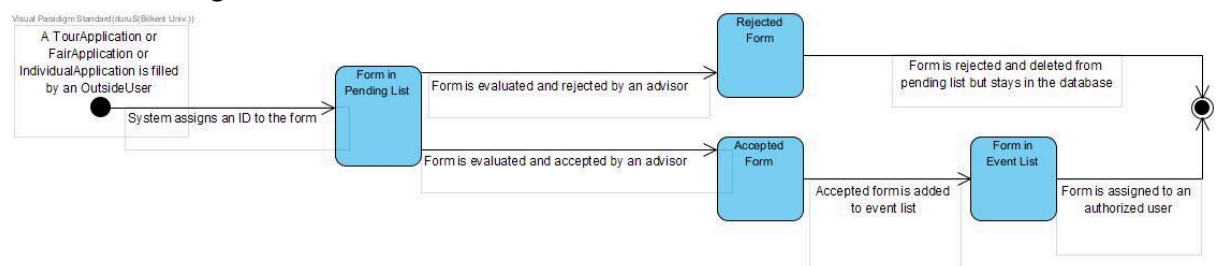## 2. State Diagrams
2.1 State Diagram for Forms

Figure 1: UML State Diagram of the Form Object in BILFO

In this state diagram, a TourApplication or FairApplication is filled and submitted to the system. When submitted, the form waits in the PendingList until it is evaluated by an advisor. The advisor may reject or accept the form. If it is accepted, an event is created with respect to it being a tour or a fair.

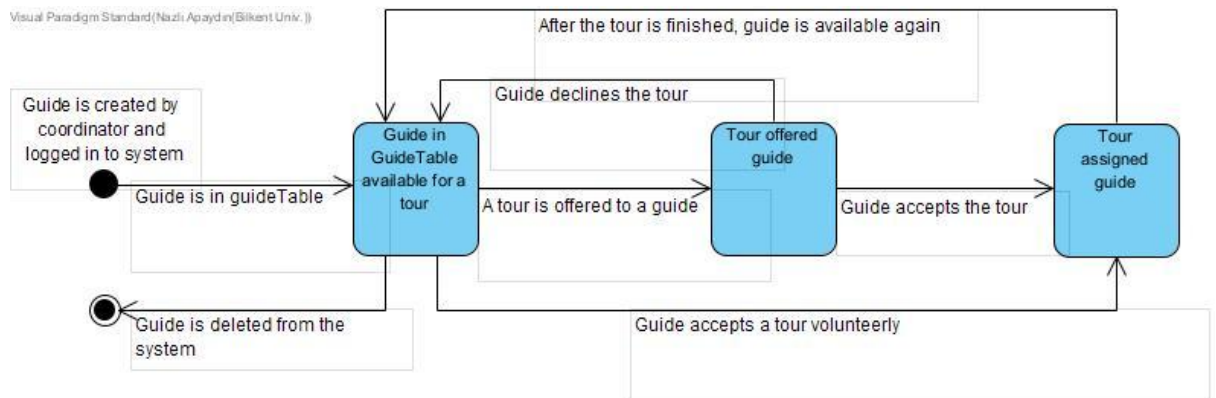## 2.2 State Diagram for Guide

Figure 2: UML State Diagram of the Guide in BILFO

In this state diagram, a guide is added to the system by admin and appears in guideTable. A guide in guideTable may either be offered a tour or may accept it voluntarily or be removed from the system. If a tour is offered, a guide can accept or decline the tour. If the tour is accepted (voluntarily or by offer), it is assigned to a guide. After the tour takes place, the guide becomes available again.

2.3 State Diagram for Events

Figure 3: UML State Diagram of the Event Object in BILFO

In this state diagram, an event is created and added to Accepted. Event can be a tour or a fair, based on its form. After the event takes place, it becomes a past event and is deleted from acceptedEvent list.

3. Activity Diagrams

3.1 Activity Diagram for Application Process
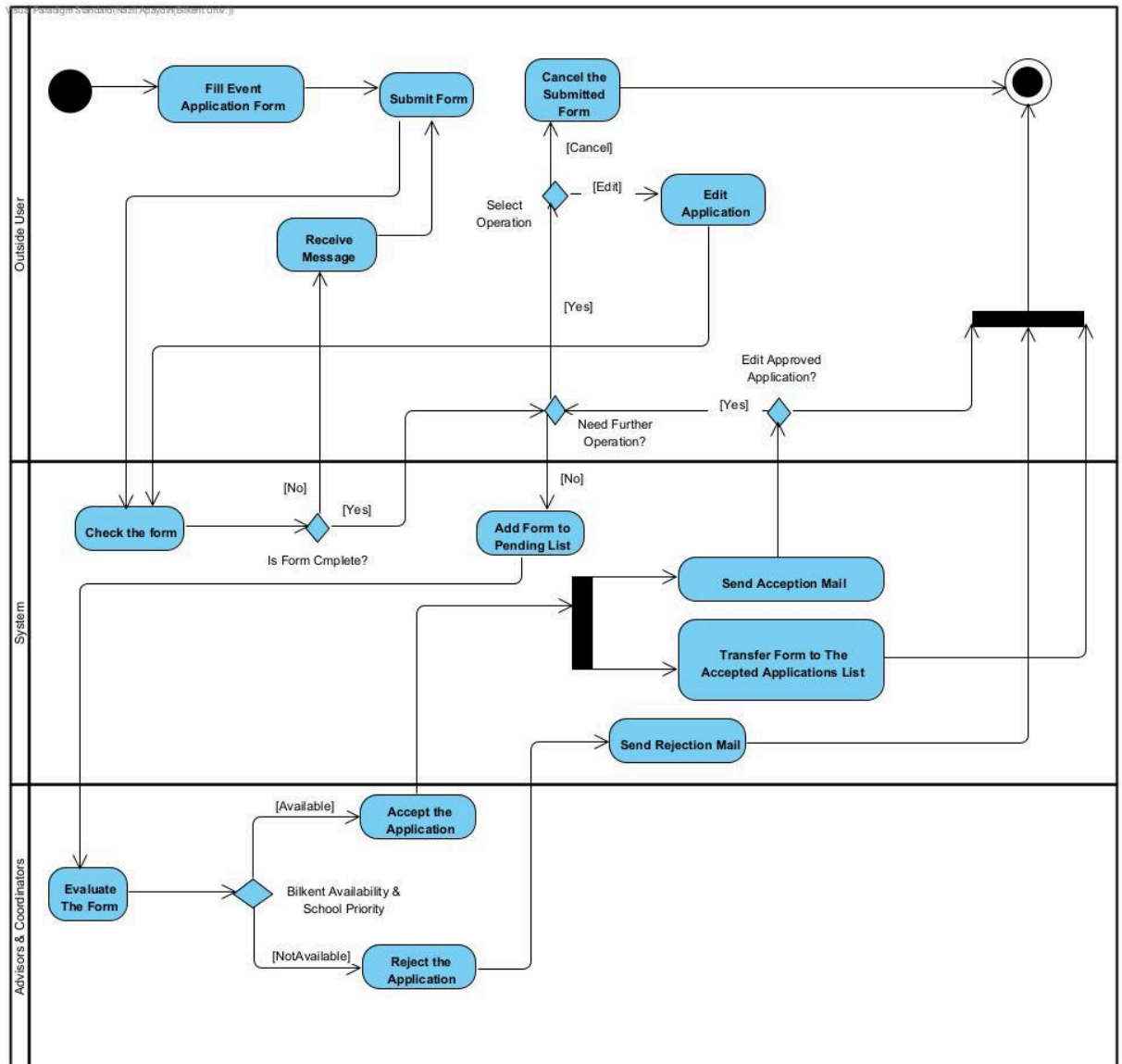
Figure 4: UML Activity Diagram of the Tour and Fair Application Process in BILFO

3.2 Activity Diagram for Accepted Tour Operations
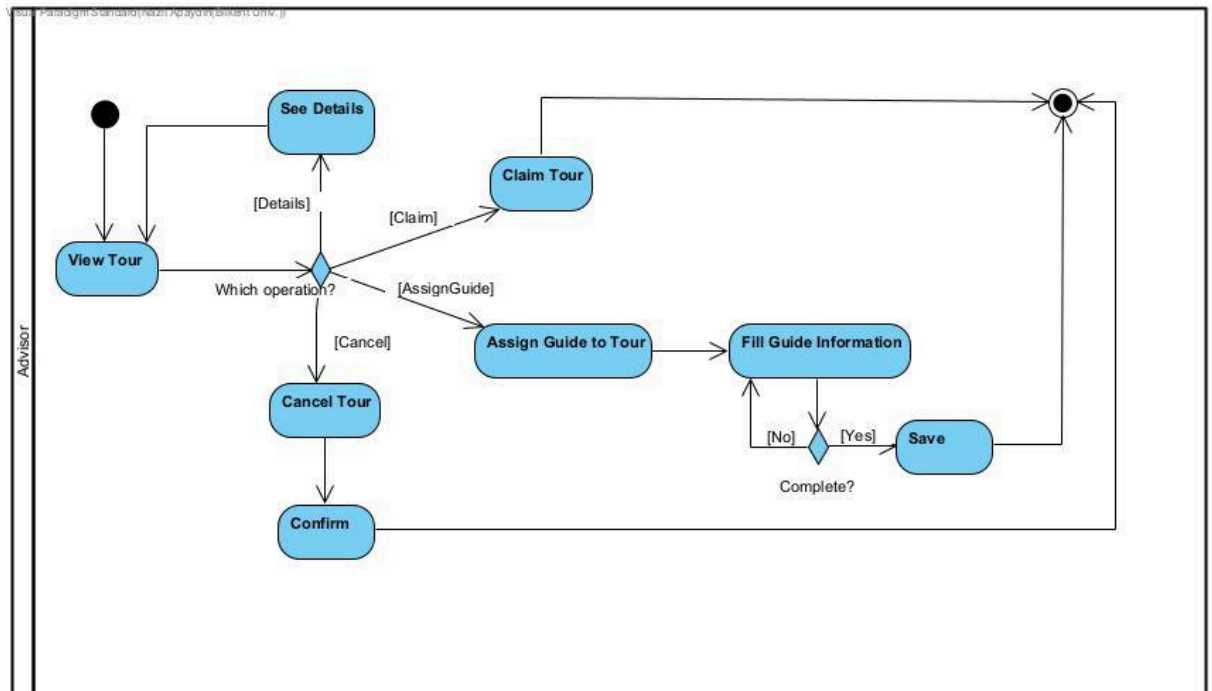
Figure 5: UML Activity Diagram of the Accepted Tour Operations in BILFO

3.3 Activity Diagram of the Guide List for Coordinators
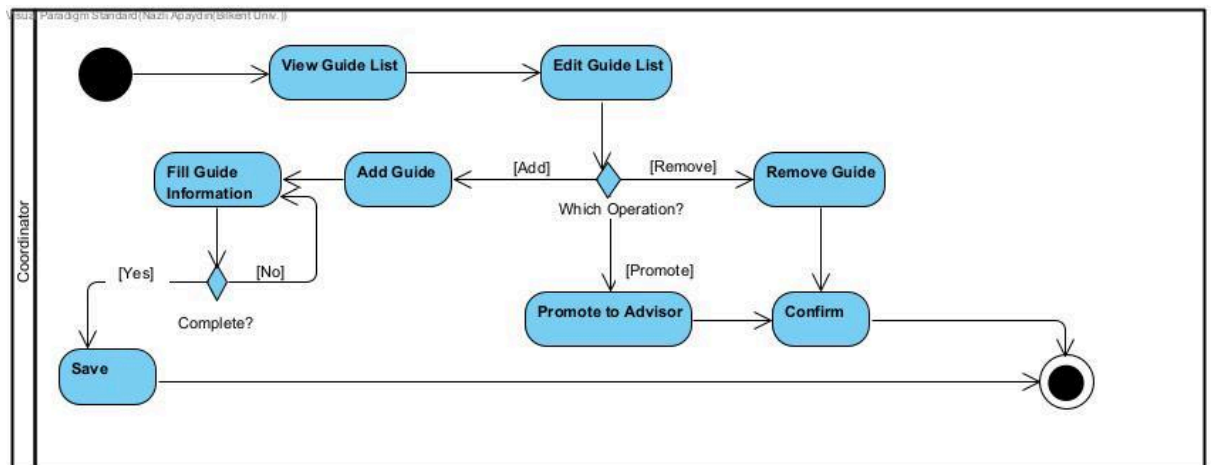
Figure 6: UML Activity Diagram of the Guide List Operations in BILFO

4. Sequence Diagrams
4.1 Forms and Events Sequence Diagram
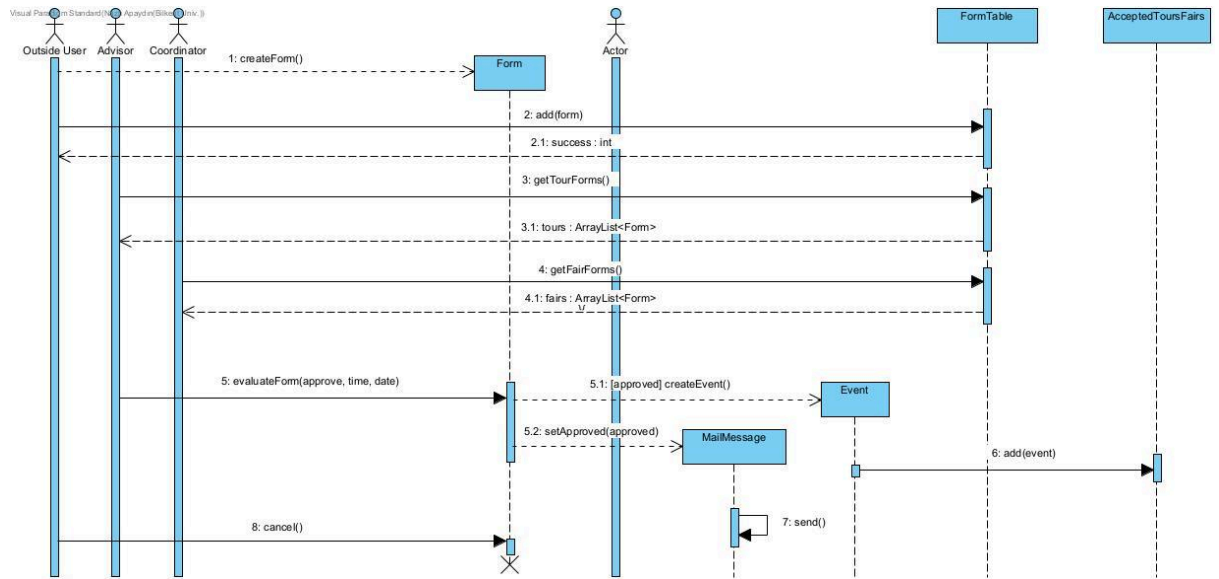
Figure 7: UML Sequence Diagram for Forms and Events

4.2 Tour Guide Assignment Sequence Diagram
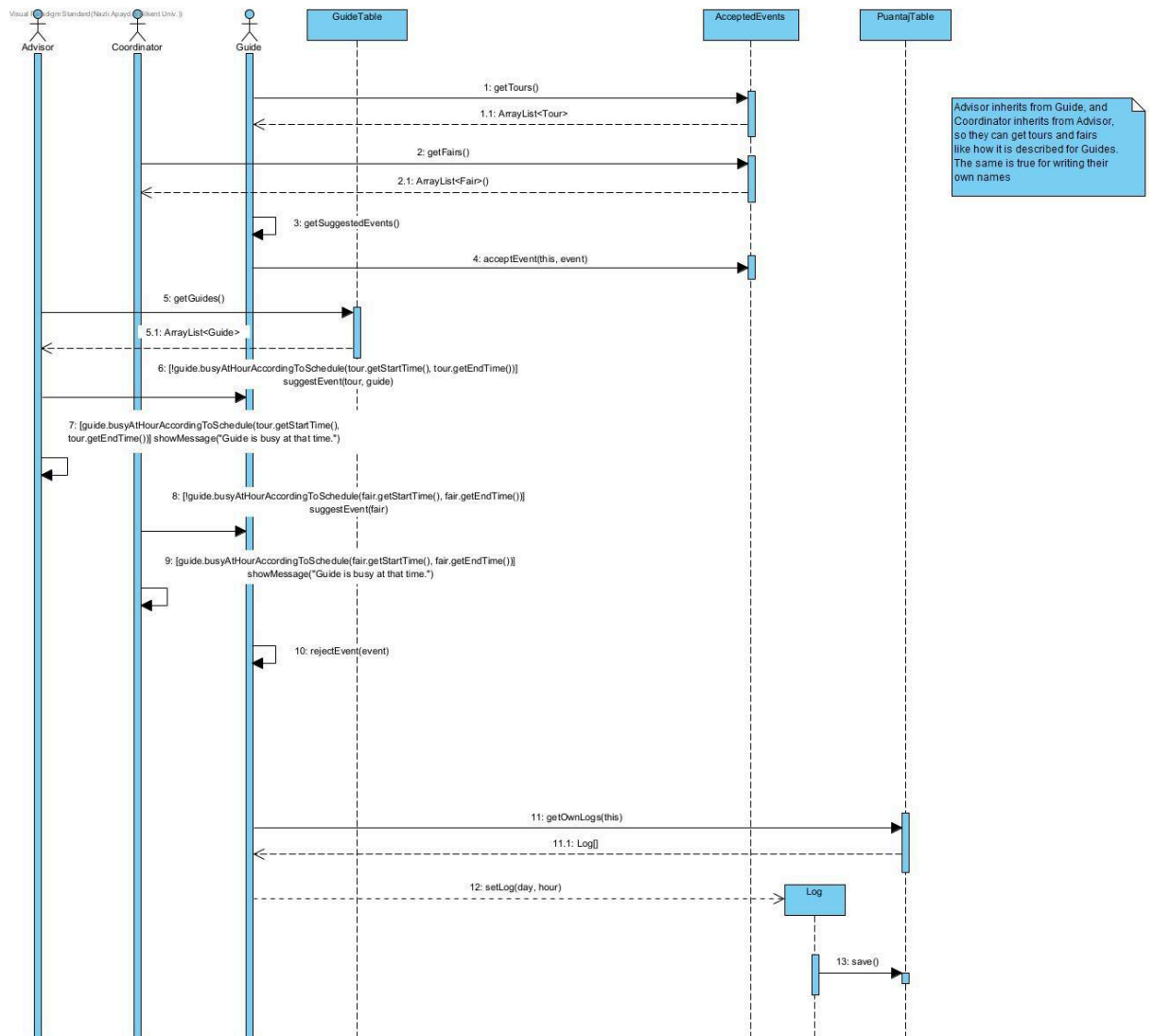
Figure 8: UML Sequence Diagram for Tour Guide Assignment

5. Class Diagram

5.1

Visual Paradigm Standard(Nazli Apaydin(Bilkent Univ.))

GuideTable
-guides : ArrayList<Guide>
+getGuides() : ArrayList<Guide>

Event
-startTime : double
-endTime : double
-date : Date
-originalForm : Form
+Event(startTime : int, endTime : int)
+getStartTime() : int
+getEndTime() : int
+isOccured() : boolean
+getOriginalForm() : Form

Fair
-guides : Guide[]
-trainees : Guide[]
+getGuides() : Guide[]
+getTrainees() : Trainee[]

Guide
-properties : GuideProperties
-schedule : bool[]
-trainee : bool
-suggestedEvents : ArrayList<Event>
+changePassword(password : String) : void
+changeHourInSchedule(hour : int, newState : bool) : void
+busyAtHourAccordingToSchedule(startTime : int, endTime : int) : bool
+getOwnLogs() : Log[]
+setLog(int day, double hour) : void
+viewAcceptedToursFairs() : AcceptedEvents
+writeOwnName(event : Event) : void
+changeLanguage() : void
+getGuideProperties() : GuideProperties
+showMessage(message : Message) : void
+isTrainee() : bool
+acceptEvent(event : Event) : void
+rejectEvent(event : Event) : void
+suggestEvent(event : Event) : void
+getSuggestedEvents() : ArrayList<Event>

AdvisorTable
-advisors : ArrayList<Advisor>
+getAdvisors() : ArrayList<Advisor>

Advisor
-dayOfAdvisor : String
+getGuideTable() : GuideTable
+addGuide(id : int, name : String) : void
+removeGuide(guide : Guide) : void
+getPendingTourList() : FormTable
+evaluateForm(form : Form, eval : boolean) : void
+getDayOfAdvisor() : String

AcceptedEvents
-events : ArrayList<Event>
+getTours() : ArrayList<Tour>
+getFairs() : ArrayList<Fair>
+getEvents() : ArrayList<Event>
+acceptEvent(guide : Guide, event : Event)

Tour
-guides : Guide[]
-trainees : Trainee[]
-visitorCount : int
+getGuides() : Guide[]
+getTrainees() : Trainee[]
+getVisitorCount() : int
+Tour(startTime : int, endTime : int, visitorCount : int)

DatabaseManager
-hashedPasswords : HashMap<int, String>
-users : ArrayList<Guide>
-forms : ArrayList<Form>
-events : ArrayList<Event>
+getActingDirector() : ActingDirector
+getCoordinator() : Coordinator
+getHashedPasswords() : HashMap<int, String>
+getForms() : ArrayList<Form>
+getEvents() : ArrayList<Event>

GuideProperties
-id : int
-name : String
-totalHoursOfService : double
+GuideProperties(id : int, name : String)
+getId() : int
+getName() : String
+getTotalHoursOfService() : double
+setTotalHoursOfService(value : double) : void

Admin
+getActingDirector() : ActingDirector
+getCoordinator() : Coordinator
+createActingDirector(adminPassword : String, id : int, name : String) : void
+removeActingDirector(adminPassword : String, current : ActingDirector) : void
+createCoordinator(adminPassword : String, id : int, name : String) : void
+removeCoordinator(adminPassword : String, current : Coordinator) : void
+changeCoordinator(adminPassword : String, current : Coordinator, advisor : Advisor) : void
+listUsers() : ArrayList<User>
+setUserProperties(adminPassword : String, guide : Guide, guideProperties : GuideProps)
+deleteUser(adminPassword : String, user : Guide)
+changeLanguage() : void
+passwordChange(code : String, guide : Guide, newPassword : String)

School
-name : String
+getName() : String
+setName(name : String) : void
+bilkentAdmissions() : int
+getSaySlore() : double
+getSozScore() : double
+getEaScore() : double
+getDilScore() : double
+calculateType() : int

Councillor
-name : String
-email : String
-phoneNo : int
+getName() : String
+setName(name : String) : void
+getEmail() : String
+setEmail(email : String) : void
+getPhoneNo() : int
+setPhoneNo(phoneNo : int) : void

ActingDirector
+createCoordinator(id : int, name : String) : void
+remove Coordinator(current : Coordinator) : void
+changeCoordinator(current : Coordinator, advisor : Advisor) : void

Coordinator
+addAdvisor(id : int, name : String) : void
+removeAdvisor(advisor : Advisor) : void
+demoteAdvisor(advisor : Advisor) : void
+promoteToAdvisor(guide : Guide) : void
+getPuanta(Table() : PuantajTable
+getAdvisor(Table() : AdvisorTable
+getDashboard() : Dashboard
+setDayOfTheAdvisor(advisor : Advisor, day : String) : void
+acceptEvent(event : Event) : void

Table
-numRow : int
-numCol : int
+getRow() : int
+getCol() : int
+add(object : Object) : int
+sort(comparator : Comparator<Object>) : void
+remove(filter : Predicate<Object>)
+refresh() : void
+remove(object : Object) : bool

FormTable
-forms : ArrayList<Form>
+getForms() : ArrayList<Form>
+getTourForms() : ArrayList<Form>
+getFairForms() : ArrayList<Form>

Dashboard
-councillors : ArrayList<Councillor>
-schools : ArrayList<School>
+getCouncillors() : ArrayList<Councillor>
+getSchools() : ArrayList<School>

school types
include:
?
Maintanence
Nurturing
*

PuantajTable
-logs : HashMap<Guide, Log[]>
+refreshTable()
+deleteLog(logID)
+getLog(logID)
+getOwnLogs(guide : Guide)

Log
-hours : double
-schoolName : String
+Log(hours : double, schoolName : String)
+getHours() : double
+setHours(hours : double) : void
+getSchoolName() : String
+setSchoolName(schoolName : String) : void
+save()

0..*

OutsideUser
+changeLanguage() : void
+login(id : int, password : String)
+createForm() : void
+saveForm() : void

Form
-id : String
-approved : boolean
-possibleTimes : HashMap<Date, double>
+cancel()
+setApproved(approved) : void
+getPossibleTimes() : HashMap<Date, double>
+evaluateForm(approve : bool, time : double, date : Date)

FeedbackForm
-guideId

Message
-title : String
-content : String
+send()
+Message(title : String, content : String)

MailMessage
-adress : String
+MailMessage(title : String, content : String, mail : String)

IndividualApplication
-visitorCount

FairApplication

HSTourApplication
-visitorCount

Figure 9: UML Class Diagram for BILFO

8

6. Mockups
6.1 Authentication and Home Page

Figure 10: Mockup for Log In Page

Figure 11: Mockup for Forgot Password

Figure 12: Mockup for Forgot Password- Setting New Password

Figure 13: Mockup for Home Page for Outside Users.

6.2 Profile and Dashboard
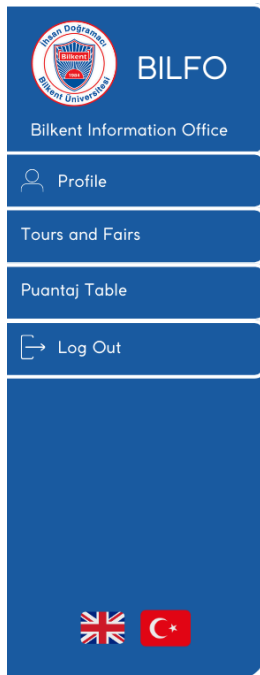
Figure 14: Mockup for Seeing Own Profile
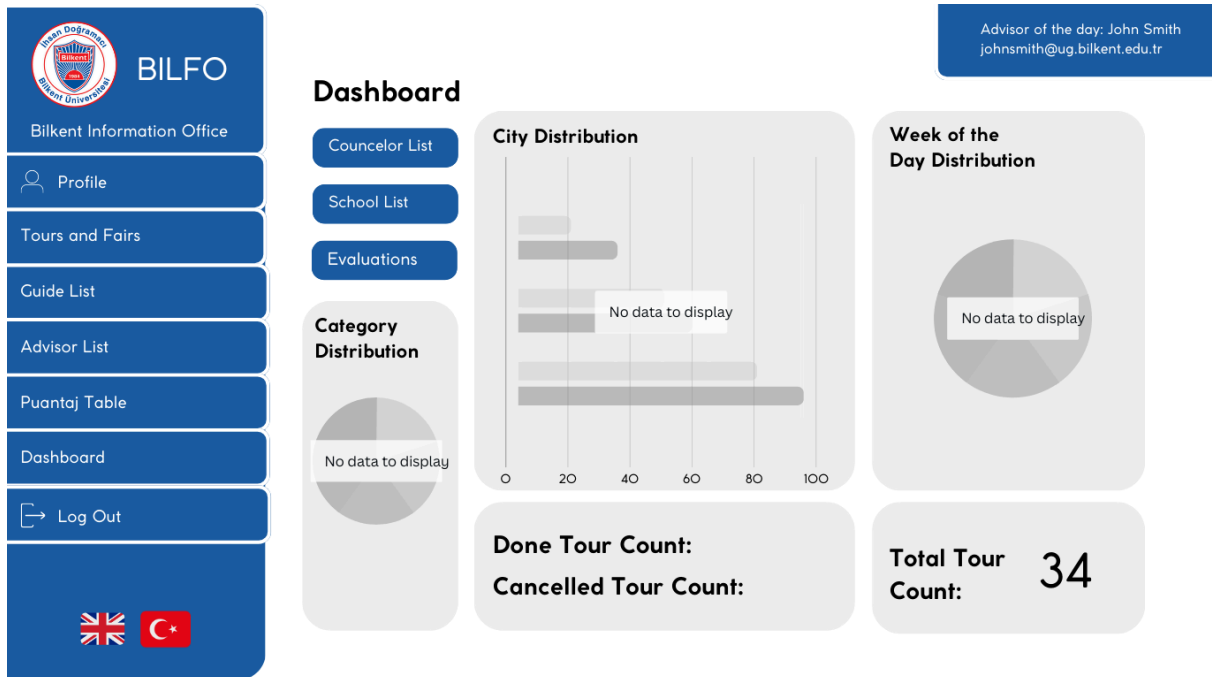
Figure 15: Mockup for Editing Own Profile

Figure 16: Mockup for Coordinator and Acting Director's Dashboard for stats.

Advisor of the day: John Smith
johnsmith@ug.bilkent.edu.tr

Counselor List

Name	School	E-mail	Phone	
Can Simit	Ankara Fen Lisesi	cansimiit@gmail.com	O5xx xxx xx xx	Remove
Mehmet Yılmaz	İzmir Fen Lisesi	mehmetyilmaz@gmail.com	O5xx xxx xx xx	Remove
Ali Öztürk	Ted Ankara Koleji	aliöztürk@gmail.com	O5xx xxx xx xx	Remove
Fatma Aydın	Tofaş Fen Lisesi	fatmaaydin@gmail.com	O5xx xxx xx xx	Remove
Hasan Yıldız	Bilkent Erzurum Lisesi	hasanyildiz@hotmail.com	O5xx xxx xx xx	Remove

Profile
Tours and Fairs
Guide List
Advisor List
Puantaj Table
Dashboard
Log Out

Figure 17: Mockup for Seeing Counselor List

BILFO

Bilkent Information Office

Advisor of the day: John Smith
johnsmith@ug.bilkent.edu.tr

School List

School Name	SAY	EA	SOZ	DIL	Bilkent Admissions	Category	
Ankara Fen Lisesi	XXX	XXX	XXX	XXX	XXX	XXX	See Previous Tours
İzmir Fen Lisesi	XXX	XXX	XXX	XXX	XXX	XXX	See Previous Tours
Ted Ankara Koleji	XXX	XXX	XXX	XXX	XXX	XXX	See Previous Tours
Tofaş Fen Lisesi	XXX	XXX	XXX	XXX	XXX	XXX	See Previous Tours
Bilkent Erzurum Lisesi	XXX	XXX	XXX	XXX	XXX	XXX	See Previous Tours

Profile
Tours and Fairs
Guide List
Advisor List
Puantaj Table
Dashboard
Log Out

Figure 18: Mockup for Seeing School List

Figure 19: Mockup for Seeing Previous Tours of a School

Figure 20: Mockup for Seeing Own Tour Schedule

Availability Schedule

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
8.20 – 9.20							
9.20 – 10.20	■						
10.20 – 11.20	■				■		■
11.20 – 12.20	■				■		■
12.20 – 13.20							■
13.20 – 14.20			■				
14.20 – 15.20			■				
15.20 – 16.20			■				
16.20 – 17.20			■				
17.20 – 18.20							
18.20 – 19.20							
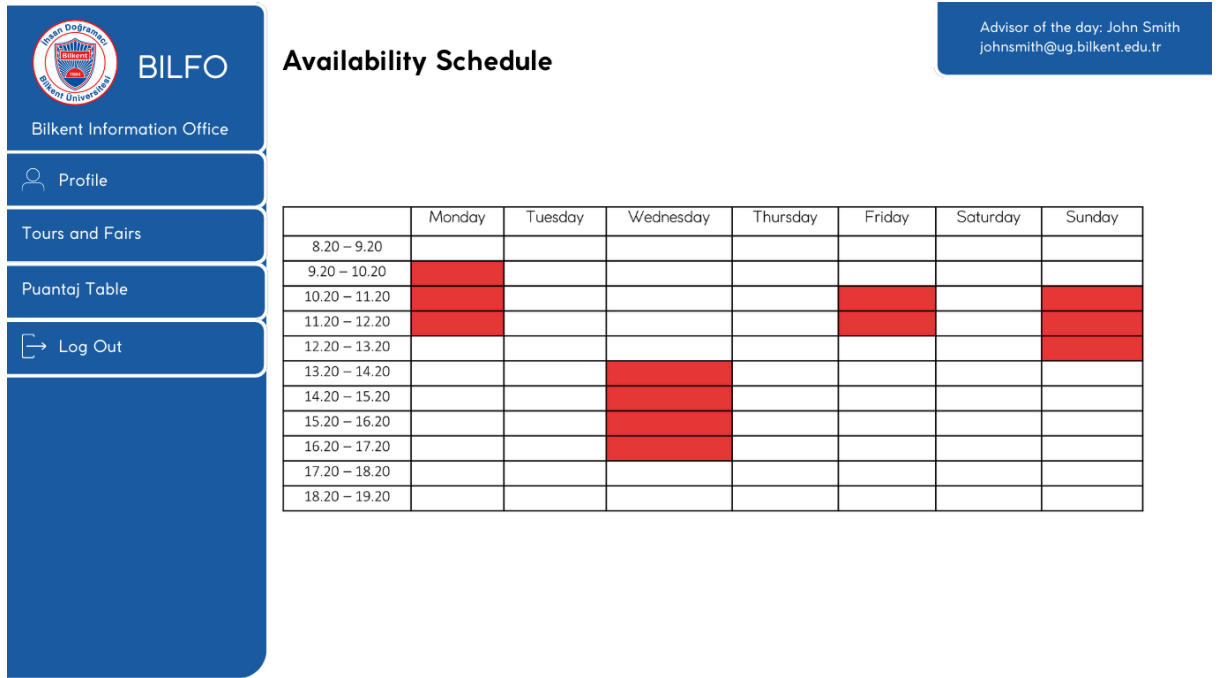
Figure 21: Mockup for Availability Schedule

6.3 Management

User List

Name	User Type	E-mail	ID		
					Add User
Nazlı Apaydın	Guide	nazliapaydin@ug.bilkent.edu.tr	22222222	Edit	Remove
Ata Uzay Kuzey	Guide	atauzaykuzey@ug.bilkent.edu.tr	22222223	Edit	Remove
Duru Solakoğlu	Guide	durusolakoglu@ug.bilkent.edu.tr	22222224	Edit	Remove
Emir Görgülü	Guide	emirgorgulu@ug.bilkent.edu.tr	22222225	Edit	Remove
Kerem Demirören	Guide	keremdemioren@ug.bilkent.edu.tr	22222226	Edit	Remove
John Smith	Advisor	johnsmith@ug.bilkent.edu.tr	22222227	Edit	Remove
Mr. Brown	Coordinator	mr.brown@ug.bilkent.edu.tr	22222228	Edit	Remove
Özhan Örge	Acting D.	orsan@bilkent.edu.tr	22222229	Edit	Remove

Figure 22: Mockup for Seeing User List - Admin View
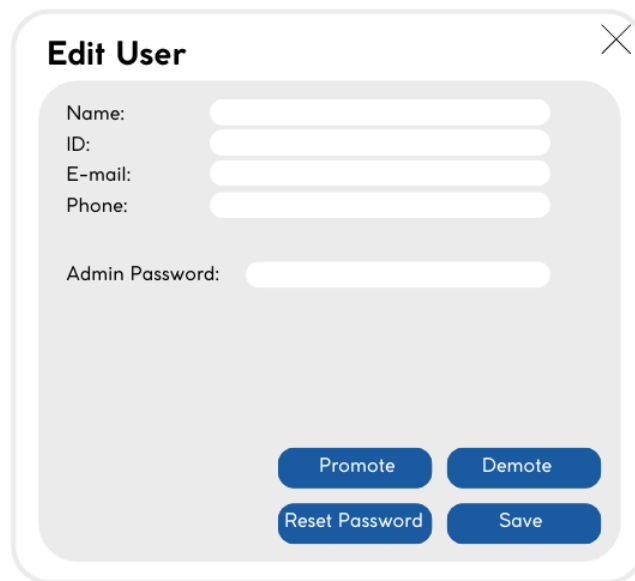
Figure 23: Mockup for Adding New User - Admin View

Figure 24: Mockup for Editing Existing User - Admin View

6.4 List and Tables

Figure 25: Mockup for Pending Tours, Advisor or Coordinator Perspective

Figure 26: Mockup for Accepted Tours, Guide Perspective (Fairs will be listed similarly, just for coordinators). If a guide suggests an event to a guide, they can choose to list the suggested events too.
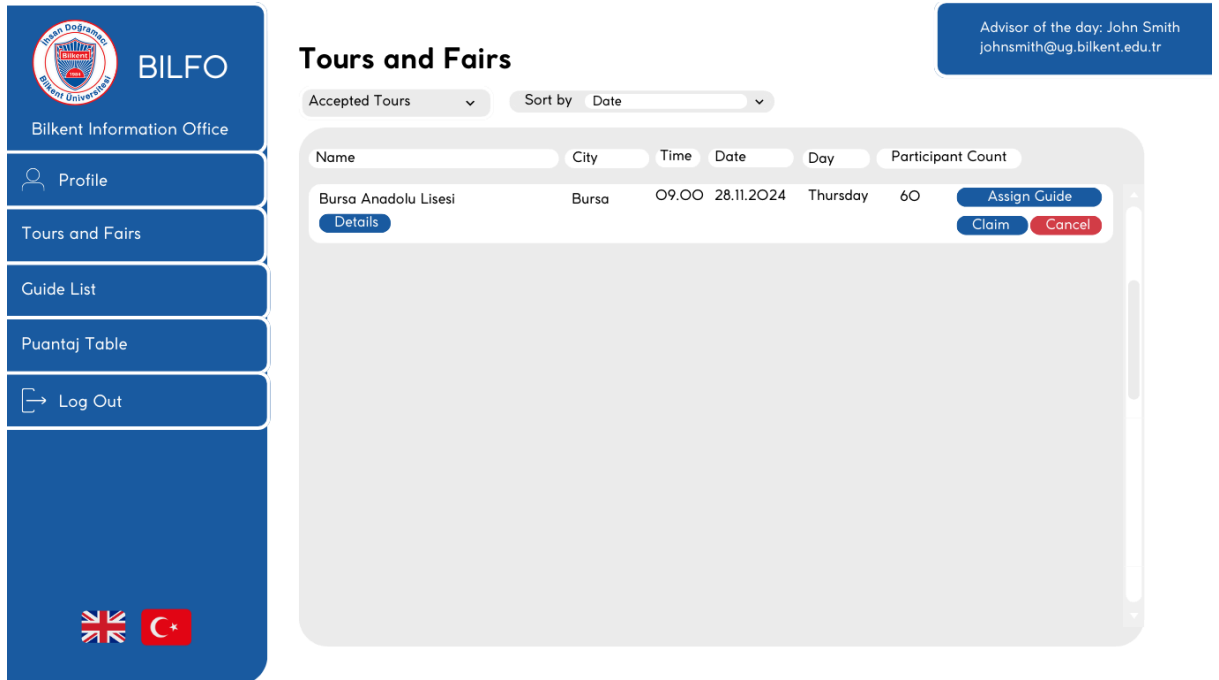
Figure 27: Mockup for Accepted Tours, Advisor Perspective
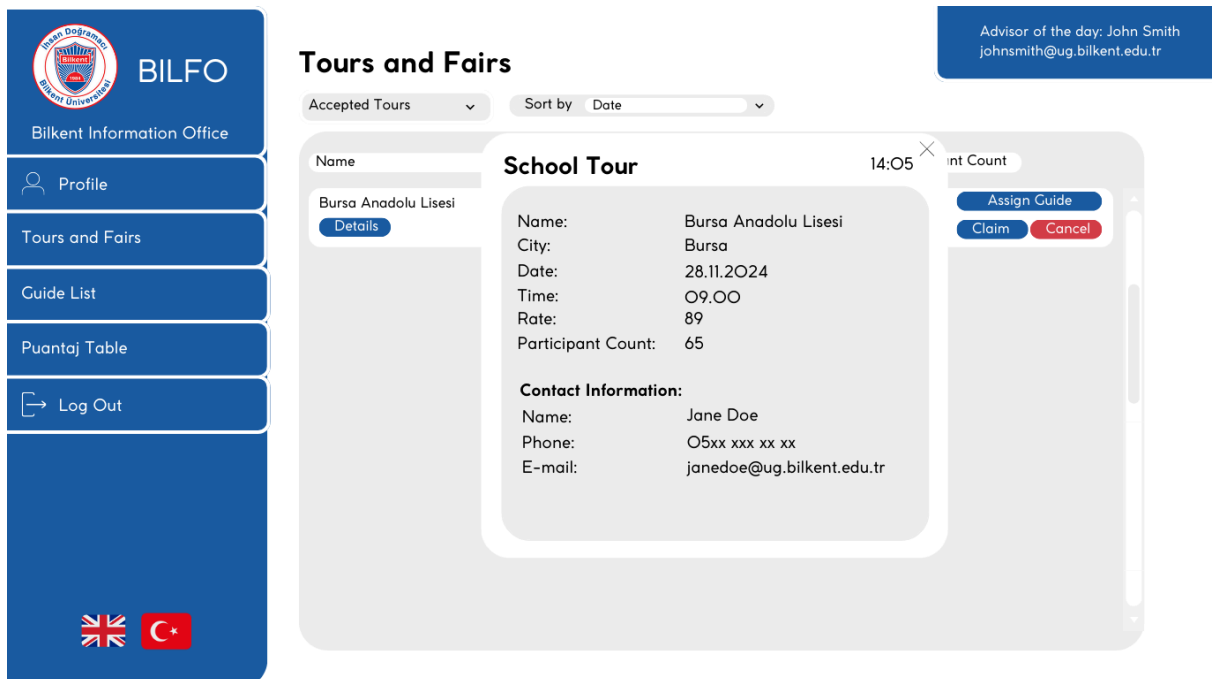
Figure 28: Mockup for Details of Accepted Tours, Advisor Perspective
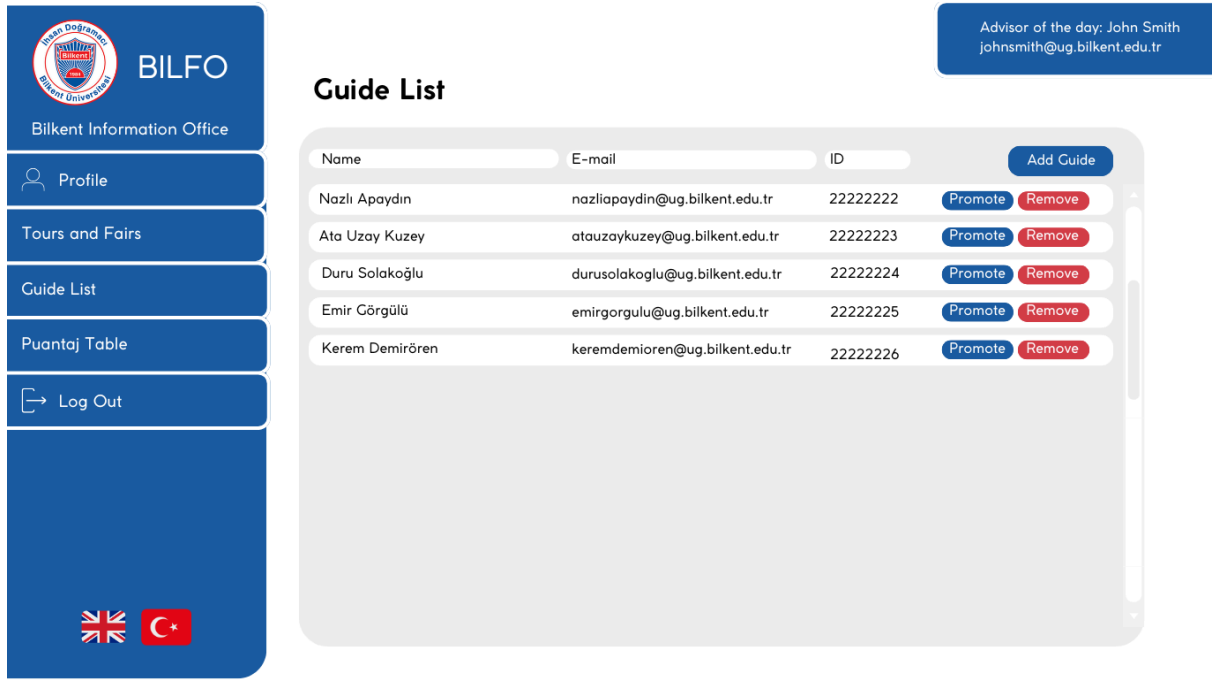
Figure 29: Mockup for Guide List, Advisor Perspective
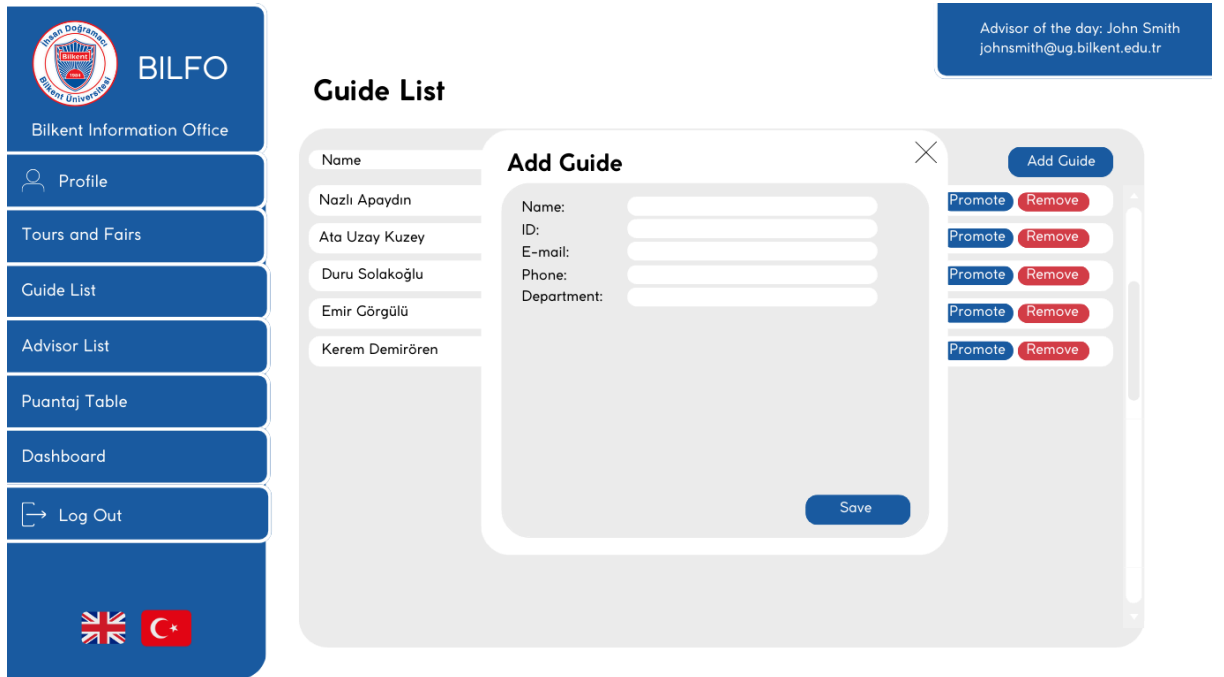
Figure 30: Mockup for Add Guide to Guide List Pop-up, Coordinator Perspective

18
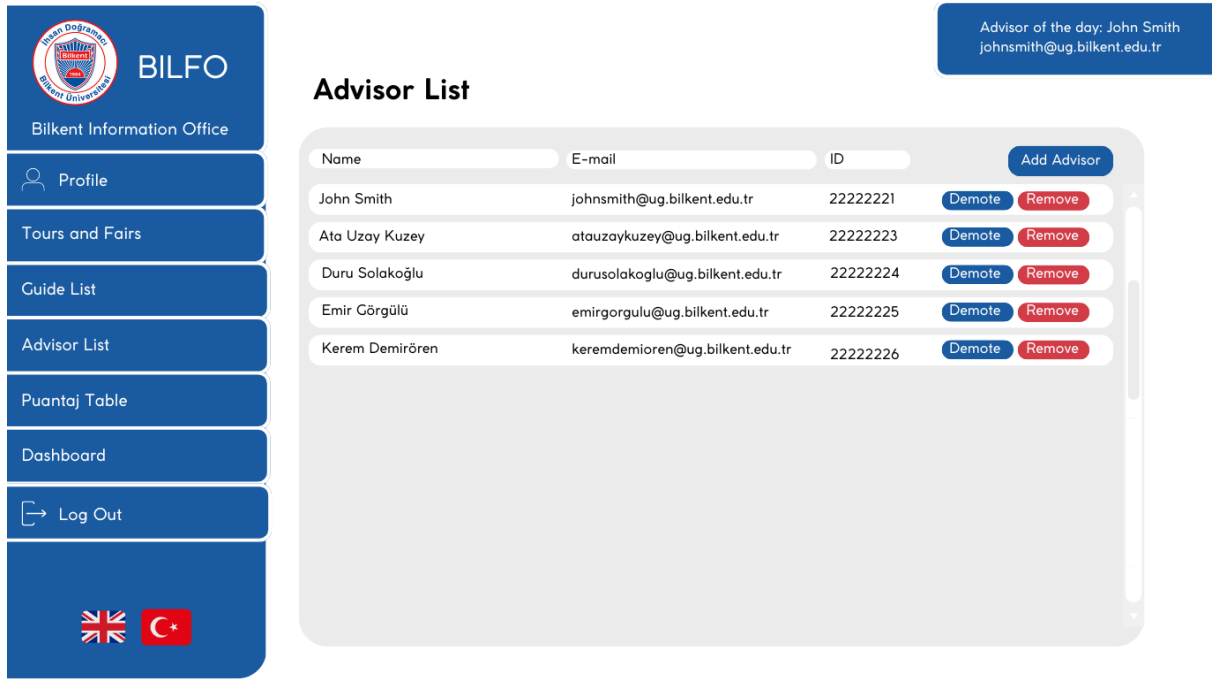
Figure 31: Mockup for Advisor List, Coordinator Perspective
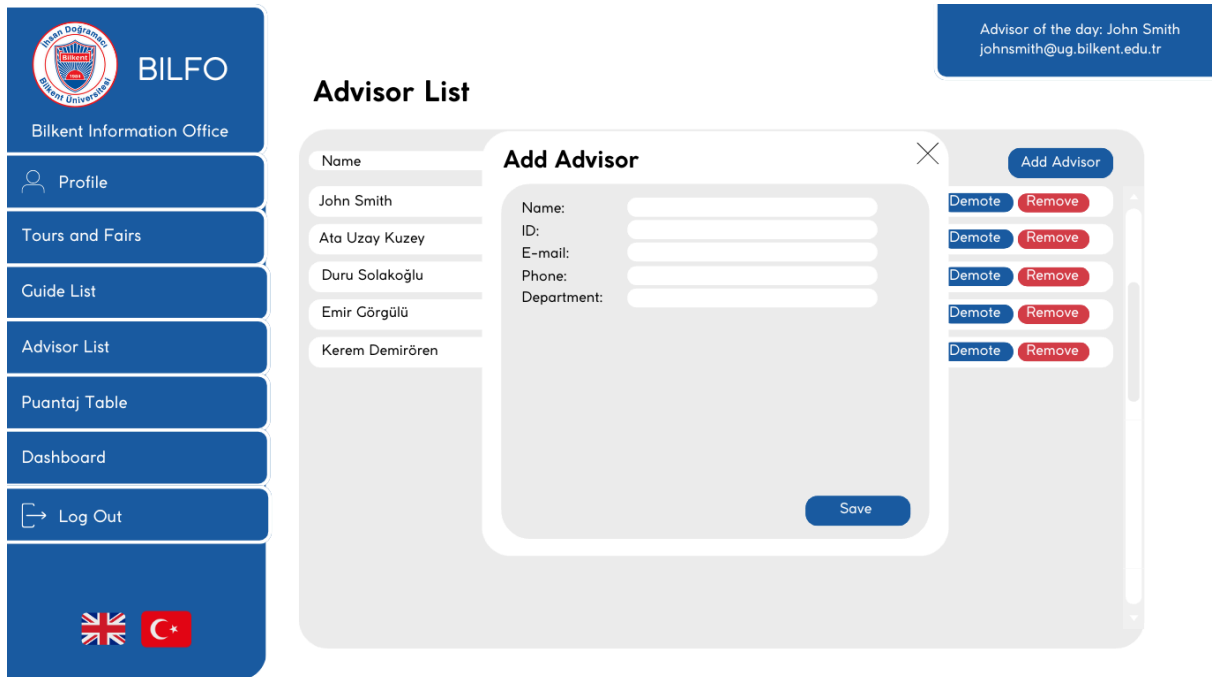
Figure 32: Mockup for Add Advisor to Advisor List Pop-up, Coordinator Perspective

19

BILFO
Bilkent Information Office

Profile

Tours and Fairs

Puantaj Table

Log Out

My Logs

Name	Type	Date	Hour	Add Log	
Bursa Anadolu Lisesi	School Tour	28.11.2024	2.5		Delete
Samsun Anadolu Lisesi	Fair	29.11.2024	2		Delete
İzmir Fen Lisesi	Individual Tour	02.12.2024	1.5		Delete
Ted Ankara Koleji	School Tour	04.12.2024	3		Delete
Atatürk Lisesi	School Tour	06.12.2024	1		Delete
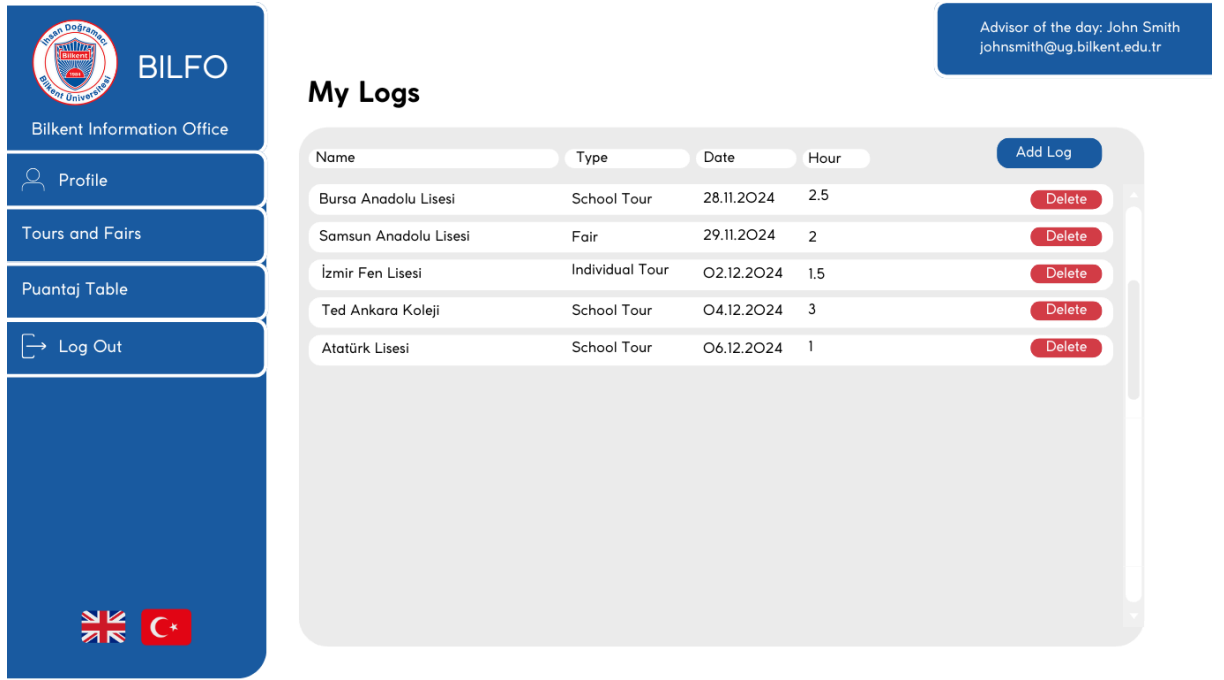
Figure 33: Mockup for Seeing Own Logs, Guide & Advisor Perspective

BILFO
Bilkent Information Office

Profile

Tours and Fairs

Puantaj Table

Log Out

My Logs

Name		Add Log
Bursa Anadolu Lisesi		Delete
Samsun Anadolu Lisesi		Delete
İzmir Fen Lisesi		Delete
Ted Ankara Koleji		Delete
Atatürk Lisesi		Delete

Add Log

Tour Name:
Date:
Hours:
Please write decimal points with ".".
Type: School Tour

Save
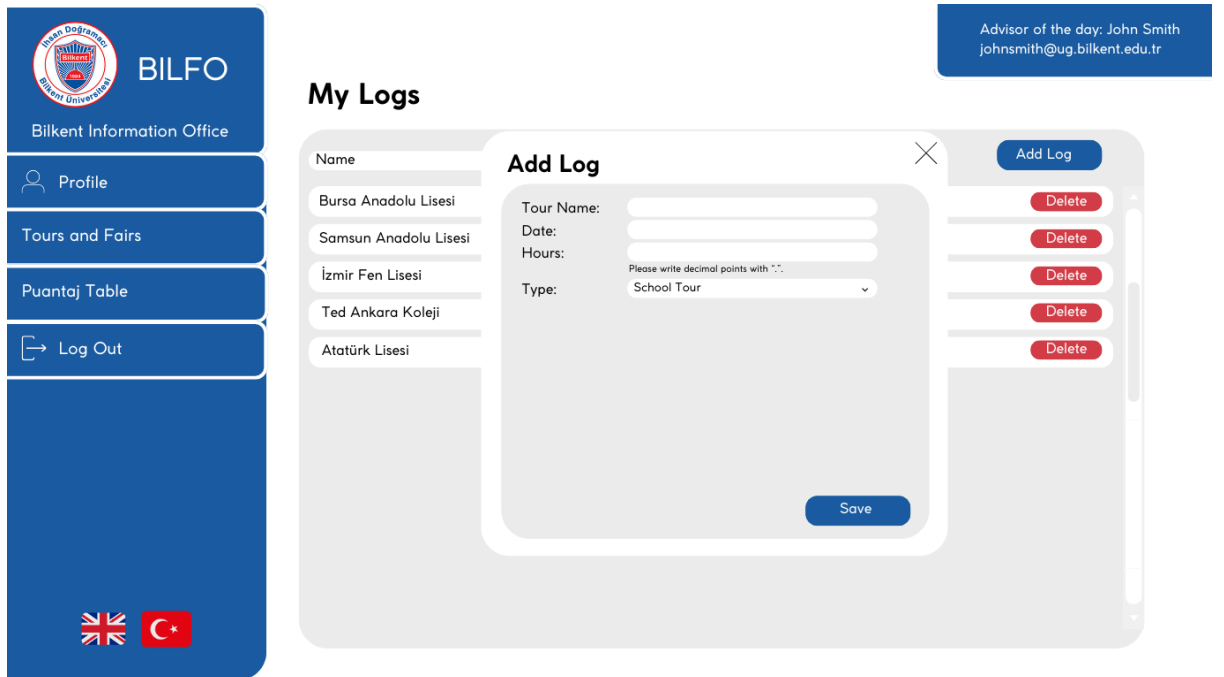
Figure 34: Mockup for Adding a Log Pop-up, Guide & Advisor Perspective

20

Figure 35: Mockup for Viewing Guides' Logs, Coordinator & Advisor Perspective

6.5 Tour and Fair Applications

Figure 36: Mockup for Evaluating a Tour
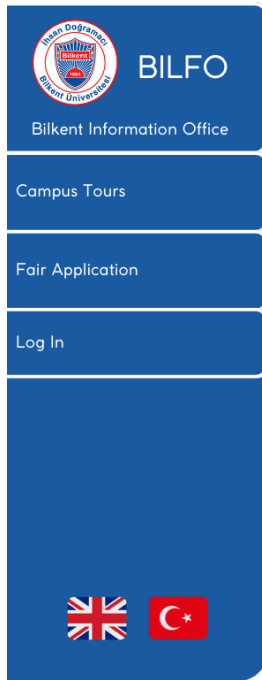
Figure 37: Mockup for Submitting a Fair Application Form

Figure 38: Mockup for Submitting a Tour Application Form

Figure 39: Mockup for Submitting an Individual Tour Application Form

📝 Evaluate Tour Application

BILFO
Bilkent Information Office

- Profile
- Tours and Fairs
- Guide List
- Puantaj Table
- Log Out
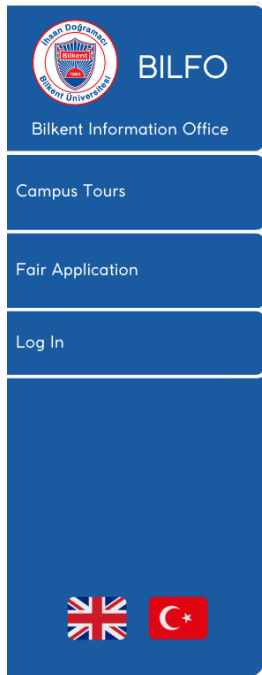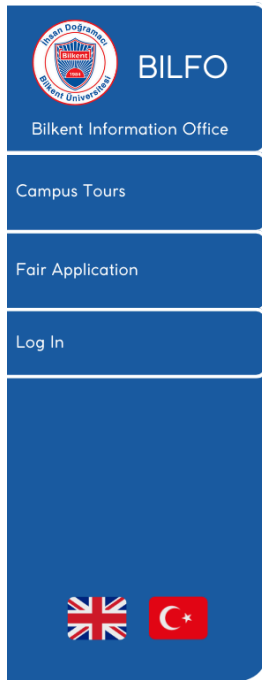
Name of the school:
Asal Kolej

City:
Ankara

First preference:
December 19th (Tuesday) - 11.00 ⬤

Second preference:
December 19th (Tuesday) - 13.30 ◯

Third preference:
December 20th (Wednesday) - 11.00 ◯

Number of visitors:
50

Counsellor
Boray Guvenc

Role of the counsellor:
School counsellor

Counsellor's phone number:
0 509 174 0927

Counsellor's e-mail:
bestofboray@gmail.com

Visitor notes:
We have two students who use wheelchairs; please take this into consideration!

Before you ACCEPT, write the number of guides needed: ▢

Accept

Before you DECLINE, explain the reason:

Decline

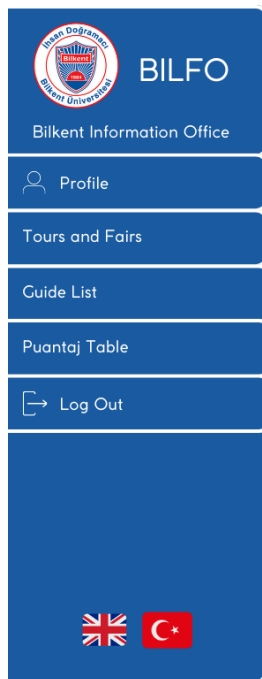Figure 40: Mockup for Evaluating a Tour Application Form
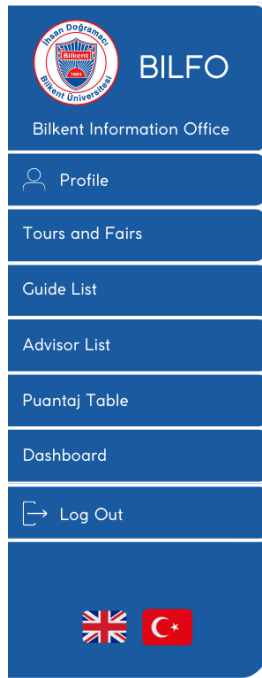
Figure 41: Mockup for Evaluating a Fair Application by Coordinator

Figure 42: Mockup for Evaluating an Individual Tour Application by an Advisor
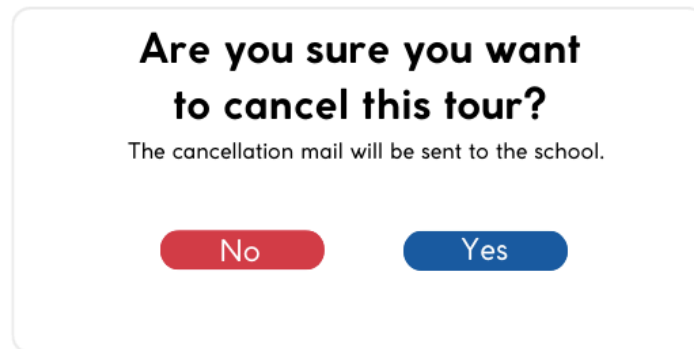
6.6 Pop-Ups

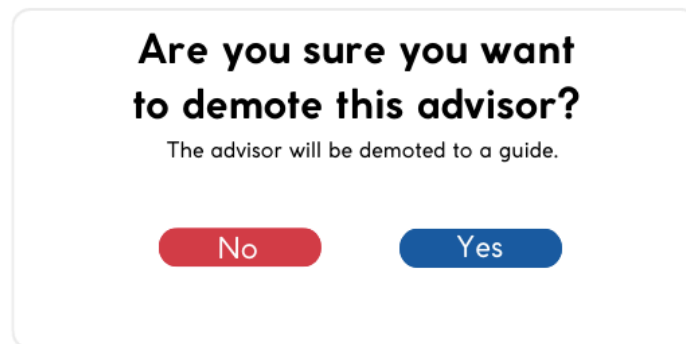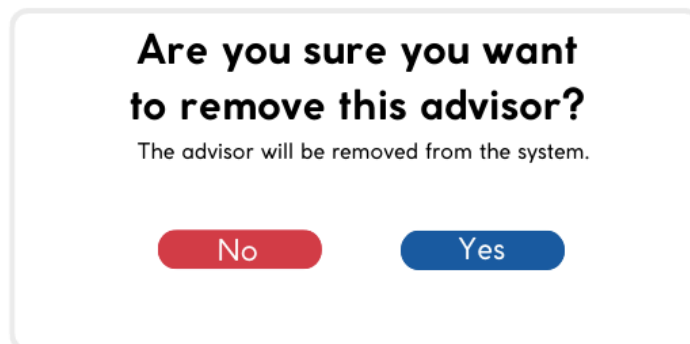Figure 43: Mockup for Cancel Tour Pop-up

Figure 44: Mockup for Demote Advisor Pop-up

Figure 45: Mockup for Remove Advisor Pop-up