



CS319 Term Project

Deliverable 1 - Final

CS319 Section 1

Sami Bora Akoğuz 22202184

Ege Ertem 22202433

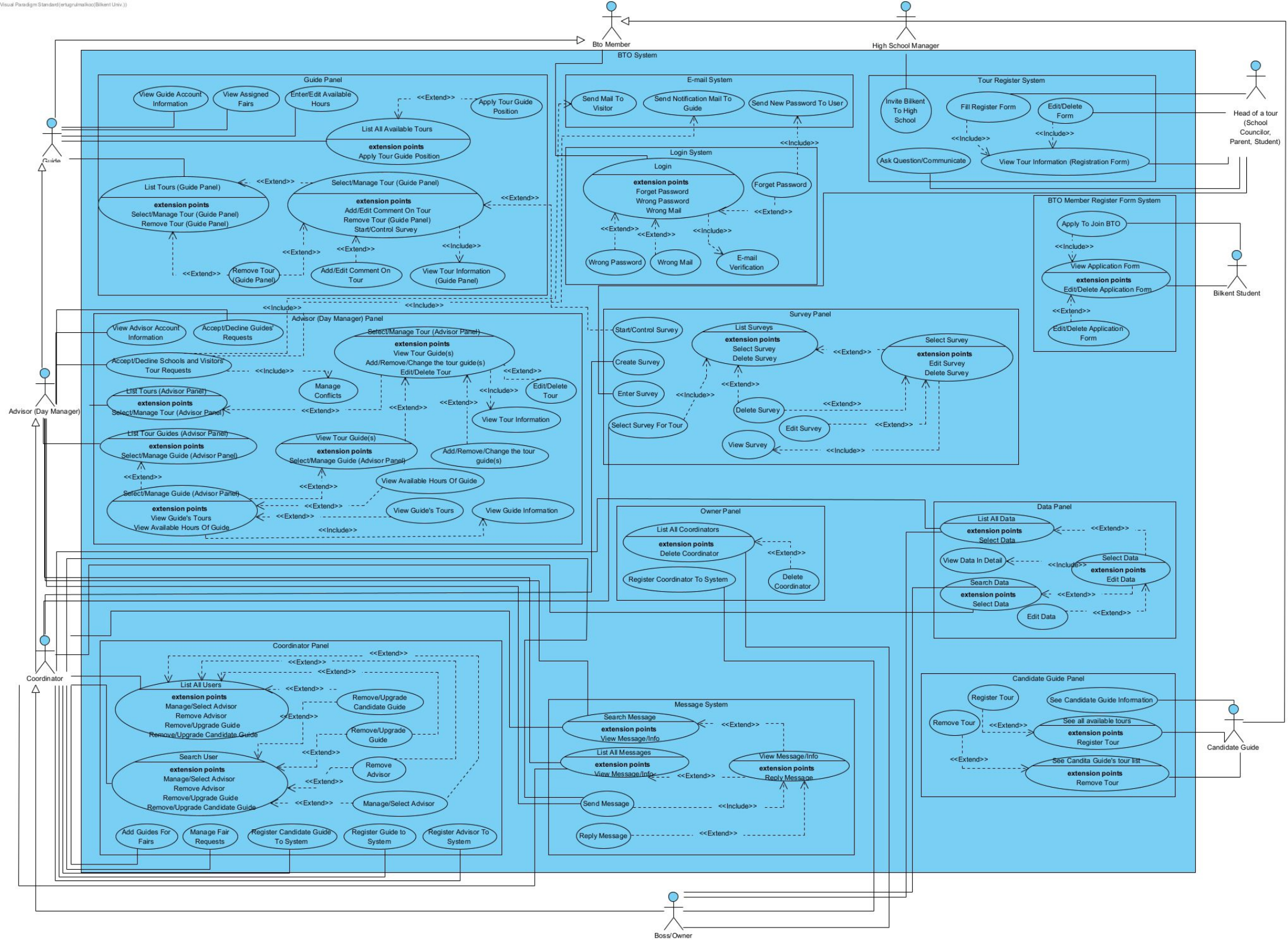
Can Kütükoğlu 22202619

Kerem Cindaruk 22201907

Ertuğrul Malkoç 22102737

12 November 2024

1. Use Case Diagram.....	2
2. Use Case Diagram Textual Descriptions.....	2
3. Tech Stack.....	22
3.1 ASP.NET.....	22
3.2 React.js.....	23
3.3 PostgreSQL.....	23
3.4 Docker.....	23
3.5 Github.....	23
4. References.....	24



2. Use Case Diagram Textual Descriptions

1. Name: Invite Bilkent to High School
 2. Participating Actor: High School Manager
 3. Entry Condition: Entering the website with a web browser, then pressing invite Bilkent button
 4. Exit Condition: Pressing the exit button on the page
 5. Flow of Events:
 - 5.1. The user presses the invite Bilkent button
 - 5.2. User confirms that he/she is a highschool coordinator
 - 5.3. After the confirmation the user is informed whether the action was successful
-
1. Name: Fill Register Form
 2. Participating Actor: Head of a Tour
 3. Entry Condition: Entering the website with a web browser, then pressing fill register form button
 4. Exit Condition: Pressing the exit button on the page
 5. Flow of Events:
 - 5.1. The user presses the fill register form button
 - 5.2. User fills the displayed registration form, according to the purpose and features of the tour
 - 5.3. User selects an appropriate date and time from the displayed table for the tour
 - 5.4. After confirming the tour application, a code will be sent to the user's mail, in order for the user to manage, edit and cancel the tour
 - 5.5. The user will be notified whether the action was successful
-
1. Name: View Tour Information
 2. Participating Actor: Head of a Tour
 3. Entry Condition: Entering the website with a web browser, having filled a registration form and receiving a code, pressing the view tour information button
 4. Exit Condition: Pressing the exit button on the page or canceling the tour
 5. Flow of Events:
 - 5.1. The user will enter the code of the tour to the displayed text box
 - 5.2. If the code is invalid
 - 5.2.1. The user will be given an error
 - 5.3. If the code matches successfully
 - 5.3.1. User's tour information will be displayed on the page
 - 5.3.2. The user will be able to ask questions to the advisors and edit or cancel the tour
 - 5.3.3. If the user presses cancel button
 - 5.3.3.1. The tour will be canceled, the user will be notified and directed to the previous screen
-
1. Name: Ask Questions / Communicate
 2. Participating Actor: Head of a Tour

3. Entry Condition: Entering the website with a web browser, having filled a registration form and received a code, pressing the view tour information -> ask questions button
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. The user presses the ask questions button
 - 5.2. The page will allow users to view and reply messages
 - 5.3. If user presses reply message
 - 5.3.1. A text box will appear where user can reply a specific message

1. Name: Enter Survey
2. Participating Actor: Head of a Tour
3. Entry Condition: Scanning the QR code which guide provided
4. Exit Condition: Pressing the exit button on the survey page
5. Flow of Events:
 - 5.1. The visitor scans the QR code
 - 5.2. The directed page will allow users to view and fill the survey

1. Name: Edit/Delete Form
2. Participating Actor: Head of a Tour
3. Entry Condition: Entering the website with a web browser, having filled a registration form and received a code, pressing the view tour information -> edit/delete form button
4. Exit Condition: Pressing the exit button on the page, delete button and save button
5. Flow of Events:
 - 5.1. The user presses the edit/delete button
 - 5.2. The page will allow users to edit and delete form
 - 5.3. If user presses delete button
 - 5.3.1. The registration form is deleted
 - 5.4. If user presses save button
 - 5.4.1. Edited information are saved

1. Name: View Guide Account Information
2. Participating Actor: Guide
3. Entry Condition: The guide should be logged in to the guide panel and press the profile picture
4. Exit Condition: Pressing the exit button on the page or save button
5. Flow of Events:
 - 5.1. The user presses the profile picture
 - 5.2. The user views the info and can edit the info
 - 5.3. If the user presses save button
 - 5.3.1. Saves and quits the page

1. Name: View Assigned Fairs
2. Participating Actor: Guide
3. Entry Condition: The guide should be logged in to the guide panel and press the view fairs button
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:

- 5.1. The guide presses the view fairs button on the guide panel
- 5.2. The guide views the fairs which are assigned by the coordinator

- 1. Name: List All Available Tours
- 2. Participating Actor: Guide
- 3. Entry Condition: The guide should be logged in the guide panel and press the list all available tours button
- 4. Exit Condition: Pressing the exit button on the page
- 5. Flow of Events:
 - 5.1. The guide presses the list all available tours button on the guide panel
 - 5.2. The guide can apply for a tour from the displayed table
 - 5.2.1. The guide presses the apply button
 - 5.2.2. A message is displayed whether the action was successful

- 1. Name: Apply Tour Guide Position
- 2. Participating Actor: Guide
- 3. Entry Condition: The guide should be logged in the guide panel and press the list all available tours -> Apply
- 4. Exit Condition: Pressing the exit button on the page
- 5. Flow of Events:
 - 5.1. The guide presses the list all available tours -> apply buttons sequentially
 - 5.2. A message is displayed whether the action was successful, the tour is deleted from the page

- 1. Name: Enter/Edit Available Hours
- 2. Participating Actor: Guide
- 3. Entry Condition: The guide should be logged in the guide panel and press the enter/edit available hours button
- 4. Exit Condition: Pressing the exit button on the page
- 5. Flow of Events:
 - 5.1. The guide presses the edit available hours button on the guide panel
 - 5.2. The guide selects his/her available hours from the displayed table

- 1. Name: List Guide's Tours
- 2. Participating Actor: Guide
- 3. Entry Condition: The guide should be logged in the guide panel and press the list guide's tours button
- 4. Exit Condition: Pressing the exit button on the page
- 5. Flow of Events:
 - 5.1. The guide presses the list guide's tours button on the guide panel
 - 5.2. Applied tours are displayed at the table with select and manage buttons

- 1. Name: Select/Manage Tour
- 2. Participating Actor: Guide
- 3. Entry Condition: The guide should be logged in the guide panel and the list guide's tours page. Then, the guide should press the select/manage tour button.
- 4. Exit Condition: Pressing the exit button on the page or pressing the drop tour button
- 5. Flow of Events:

- 5.1. Tour information will be displayed on the page
- 5.2. Comment and start/control survey buttons are displayed
- 5.3. Drop tour functionality will be provided to the guide
- 5.4. If the guide presses the drop tour button
 - 5.4.1. The corresponding tour will be dropped
 - 5.4.2. The guide will be notified whether the action was successful
 - 5.4.3. The guide will be directed to the previous page

1. Name: Start/Control Survey
2. Participating Actor: Guide
3. Entry Condition: The guide should be logged in the guide panel -> select/manage tour -> start/control survey
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. User presses the start/control survey button
 - 5.2. A QR code will be displayed on the screen along with a survey, in order visitors to fill

1. Name: View Tour Information
2. Participating Actor: Guide
3. Entry Condition: The guide should be logged in the guide panel -> list guide's tours page -> select/manage tour sequentially.
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. Tour information will be displayed on the page

1. Name: Remove Tour
2. Participating Actor: Guide
3. Entry Condition: The guide should be logged in the guide panel -> list guide's tours page -> select/manage tour -> remove tour sequentially.
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. Tour will be removed from guide's tours if the remove is pressed
 - 5.2. User will be notified with a pop-up

1. Name: Add/Edit Comment On Tour
2. Participating Actor: Guide
3. Entry Condition: The guide should be logged in the guide panel -> list guide's tours page -> add/edit comment text box sequentially.
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. User presses text box
 - 5.2. User writes a comment and presses save button
 - 5.2.1. The comment is saved

1. Name: View Advisor Account Information
2. Participating Actor: Advisor
3. Entry Condition: The advisor should be logged in to the advisor panel and press the profile picture
4. Exit Condition: Pressing the exit button on the page or save button
5. Flow of Events:
 - a. The user presses the profile picture
 - b. The user views the info and can edit the info
 - c. If the user presses save button
 - i. Saves and quits the page

1. Name: Accept / Decline schools' and visitors' tour requests
2. Actor: Advisor
3. Entry Condition: Clicking the "School Request" button
4. Exit Condition: Clicking the "Exit"
5. Flow of Events:
 - 5.1. Advisor presses "Schools' Request" button
 - 5.2. Advisor selects which request to take action
 - 5.2.1. Advisor approves the request
 - 5.2.1.1. An email is sent to the person requesting the tour, which says the request is approved
 - 5.2.2. Advisor denies the request
 - 5.2.2.1. An email is sent to the person requesting the tour, which says the request is denied
 - 5.2.3. Advisor proposes a new date and time
 - 5.2.3.1. An email is sent to the person requesting the tour, proposing the new date
 - 5.3. Advisor exits the page by clicking "Exit"

1. Name: Manage/Conflicts
2. Actor: Advisor
3. Entry Condition: Clicking the "Manage Conflicts" button
4. Exit Condition: Clicking the "Exit"
5. Flow of Events:
 - 5.1. Advisor presses "Manage Conflicts" button
 - 5.2. Advisor manages conflicts because of tours arranged at the same time.
 - 5.3. User presses the save button

1. Name: Search Message
2. Actor: Advisor and Coordinator
3. Entry Condition: Clicking the "Search Message"
4. Exit Condition: Searching the messages and clicking one of them
5. Flow of Events:
 - 5.1. User presses the "Search" Button
 - 5.2. The search bar filters the messages according to the written filter

1. Name: List All Messages
2. Actor: Advisor and Coordinator

3. Entry Condition: Clicking the “List Messages Button”
 4. Exit Condition: Clicking the “Exit” Button
 5. Flow of Events:
 - 5.1. User presses the “List Messages”
 - 5.2. The messages are listed in a container which allows the user to see all of them.
-
1. Name: View Messages / Info
 2. Actor: Advisor and Coordinator
 3. Entry Condition: Clicking “Message Info” Button
 4. Exit Condition:
 5. Flow of Events:
 - 5.1. User presses “Message Info” Button
 - 5.2. Information of the message is shown to the user.
-
1. Name: Reply Message
 2. Actor: Advisor, Coordinator
 3. Entry Condition: Clicking a message sent to the user
 4. Exit Condition: Either canceling or sending the message
 5. Flow of Events:
 - 5.1. User presses “Reply Message” Button
 - 5.2. A text box will appear for advisors and coordinators to reply messages
-
1. Name: Send Message
 2. Actor: Advisor, Coordinator
 3. Entry Condition: Clicking the “Send Message” button
 4. Exit Condition: Either canceling or sending the message
 5. Flow of Events:
 - 5.1. User presses “Send Message” Button
 - 5.2. A text box will appear for advisors and coordinators to send messages
-
1. Name: Send Mail To Visitor
 2. Actor: Advisor
 3. Entry Condition: Clicking the “Send Mail” button
 4. Exit Condition: Either canceling or sending the message
 5. Flow of Events:
 - 5.1. User presses “Send Mail” Button
 - 5.2. A text box will appear for advisors to send mails, a mail will be sent automatically when the user confirms the text message.
-
1. Name: List Tours
 2. Actor: Advisor
 3. Entry Condition: Clicking the “List Tours” Button
 4. Exit Condition: Clicking the “Return To Advisor Panel” Button or clicking to one of the tours listed
 5. Flow of Events:
 - 5.1. Advisors click on the “List Tours” Button.
 - 5.2. Tours which are under the guidance of the advisor are listed.

- 5.3. Advisors can click on one of the tours to see the tour info.
- 5.4. The clicked tour directs to the manage tour section

- 1. Name: List Tour Guides
 - 2. Actor: Advisor
 - 3. Entry Condition: Clicking the "List Tour Guides" Button
 - 4. Exit Condition: Clicking the "Return To Advisor Panel" Button
 - 5. Flow of Events:
 - 5.1. Advisor opens the page
 - 5.2. Guides under the responsibility of the advisor are listed
 - 5.3. Advisor sees the info of guides
 - 5.4. Advisor can click to the guide they want to manage
 - 5.4.1. The page is directed to the guide manager
-
- 1. Name: Select/Manage Guide (Advisor Panel)
 - 2. Actor: Advisor
 - 3. Entry Condition: Clicking the "Select/Manage Guide" Button
 - 4. Exit Condition: Clicking the "Exit" Button
 - 5. Flow of Events:
 - 5.1. Advisor presses Select/Manage Guide button
 - 5.2. Guide's information is displayed
 - 5.3. Two buttons occur which are View Guide's Tours and View Available Hours Of Guide
-
- 1. Name: View Guide Information (Advisor Panel)
 - 2. Actor: Advisor
 - 3. Entry Condition: Clicking the "List Tour Guides" Button
 - 4. Exit Condition: Clicking the "Exit" Button
 - 5. Flow of Events:
 - 5.1. Advisor presses Select/Manage Guide button
 - 5.2. Guide's information is displayed
-
- 1. Name: View Guide's Tours (Advisor Panel)
 - 2. Actor: Advisor
 - 3. Entry Condition: Clicking the "View Guide's Tours" Button
 - 4. Exit Condition: Clicking the "Exit" Button
 - 5. Flow of Events:
 - 5.1. Advisor presses "View Guide's Tours" button
 - 5.2. Guide's tours are displayed
-
- 1. Name: View Available Hours Of Guide (Advisor Panel)
 - 2. Actor: Advisor
 - 3. Entry Condition: Clicking the "View Available Hours Of Guide" Button
 - 4. Exit Condition: Clicking the "Exit" Button
 - 5. Flow of Events:
 - 5.1. Advisor presses "View Available Hours Of Guide" button
 - 5.2. Guide's available hours are displayed

1. Name: Accept / Decline Guide Requests
2. Actor: Advisor
3. Entry Condition: Clicking the “Guide Requests” Button
4. Exit Condition: Clicking the “Return To Advisor Panel” Button
5. Flow of Events:
 - 5.1. Advisor clicks on to “Guide Requests” Button
 - 5.2. Requests from guides are listed
 - 5.3. Advisor selects which one to manage
 - 5.3.1. The advisor accepts the request
 - 5.3.1.1. The Guide is added to the tour and a message is sent to the Guide
 - 5.3.2. The advisor declines the request
 - 5.3.2.1. The Guide’s request is declined and a message is sent to the Guide

1. Name: Select/Manage Tour
2. Actor: Advisor
3. Entry Condition: Selecting the tour from “List Tours” section
4. Exit Condition: Clicking to see the tour info or edit the tour or clicking return
5. Flow of Events:
 - 5.1. The advisor clicks to the tour and opens this page
 - 5.2. The advisor can select to see the tour info
 - 5.2.1. The page is directed to view information
 - 5.3. Advisor can select to edit the tour
 - 5.3.1. The page is directed to edit tour page
 - 5.4. After changes, the “Save” Button is clicked, and the page is directed to the “List Tours” page.

1. Name: View Tour Information
2. Actor: Advisor
3. Entry Condition: Clicking the “View Information” in one of the tour in the Tour List
4. Exit Condition: Clicking the “Exit” Button
5. Flow of Events:
 - 5.1. Information of the tour is shown
 - 5.2. Advisor can click the “Exit” button to exit

1. Name: Add/Remove/Change the tour guide
2. Actor: Advisor
3. Entry Condition: Clicking the “Edit” in one of the tour in the Tour List
4. Exit Condition: Clicking the “Save” or “Exit” Button
5. Flow of Events:
 - 5.1. Guide of the tour is shown
 - 5.2. The advisor can select what to do
 - 5.2.1. The advisor can assign a guide
 - 5.2.2. The advisor can remove a guide
 - 5.2.3. The advisor can change the guide
 - 5.3. The advisor can save what he/she did and exit the page

1. Name: Edit / Delete Tour
 2. Actor: Advisor
 3. Entry Condition: Selecting a tour from a Tour List
 4. Exit Condition: Clicking the "Save" or "Return"
 5. Flow of Events:
 - 5.1. The advisor click the "Manage Tour" Button
 - 5.2. The advisor can edit the tour or delete the tour
-
1. Name: View tour guide(s)
 2. Actor: Advisor
 3. Entry Condition: Selecting a tour from a Tour List
 4. Exit Condition: Clicking "Return" Button or clicking the buttons to access about the guide
 5. Flow of Events:
 - 5.1. The advisor clicks the view tours of the guide
 - 5.2. The advisor clicks the view the guide information
 - 5.3. The advisor clicks to see the available hours of the guide
-
1. Name: Register Advisor to System
 2. Actor: Coordinator
 3. Entry Condition: Pressing the "Register Advisor" button
 4. Exit Condition: Pressing "Return to Homepage" button
 5. Flow of Events:
 - 5.1. The coordinator presses the "Register Advisor" button
 - 5.2. The coordinator enters the email of the advisor
 - 5.3. The email is added to the system with the "Advisor" role
-
1. Name: Register Guide to System
 2. Actor: Coordinator
 3. Entry Condition: Pressing the "Register Guide" button
 4. Exit Condition: Pressing "Return to Homepage" button
 5. Flow of Events:
 - 5.1. The coordinator presses the "Register Guide" button
 - 5.2. The coordinator enters the email of the guide
 - 5.3. The email is added to the system with the "Guide" role
-
1. Name: Register Candidate Guide to System
 2. Actor: Coordinator
 3. Entry Condition: Pressing the "Register Candidate Guide" button
 4. Exit Condition: Pressing "Return to Homepage" button
 5. Flow of Events:
 - 5.1. The coordinator presses the "Register Candidate Guide" button
 - 5.2. The coordinator enters the email of the candidate guide
 - 5.3. The email is added to the system with the "Candidate Guide" role

1. Name: Create Survey
 2. Actor: Coordinator
 3. Entry Condition: Pressing the “Create Survey” button
 4. Exit Condition: Pressing to the button “Return to Survey Panel”
 5. Flow of Events:
 - 5.1. The coordinator presses the “Create Survey” button
 - 5.1.1. The coordinator can create a new survey for the system
-
1. Name: List Surveys
 2. Actor: Coordinator
 3. Entry Condition: Pressing the “List Surveys” button
 4. Exit Condition: Pressing the button “Return to Survey Panel”
 5. Flow of Events:
 - 5.1. The coordinator presses the “List Surveys” button
 - 5.1.1. System lists all the surveys available for use
-
1. Name: Select Survey For Tour
 2. Actor: Coordinator
 3. Entry Condition: Pressing the button “Select Survey For Tour” next to the survey they want to select for tour.
 4. Exit Condition: Pressing the button “Return to Survey Panel”
 5. Flow of Events:
 - 5.1. The coordinator presses the “Select Survey For Tour” button next to the survey they want to select from the list.
 - 5.1.1. The survey is assigned for the tour
-
1. Name: Select Survey
 2. Actor: Coordinator
 3. Entry Condition: Pressing the button “Select Survey” next to the survey they want to select.
 4. Exit Condition: Pressing the button “Return to List Surveys”
 5. Flow of Events:
 - 5.1. The coordinator presses the “Select Survey” button next to the survey they want to select from the list.
 - 5.1.1. The buttons “View Survey” and “Edit Survey” are displayed
-
1. Name: Delete Survey
 2. Actor: Coordinator
 3. Entry Condition: Pressing the button “Delete Survey” in the listed survey items or selected survey.
 4. Exit Condition: Pressing the button “Return to List Surveys”
 5. Flow of Events:
 - 5.1. The coordinator presses the “Delete Survey” next to the selected list item that is intended to be deleted.
 - 5.1.1. The Survey is deleted
 - 5.2. The coordinator presses the “Delete Survey” int the selected survey panel
 - 5.2.1. The Survey is deleted

1. Name: Edit Survey
2. Actor: Coordinator
3. Entry Condition: Pressing the button “Edit Survey” on the selected survey.
4. Exit Condition: Pressing the button “Return to List Surveys”
5. Flow of Events:
 - 5.1. The coordinator presses the “Edit Survey” button on the selected survey panel
 - 5.1.1. The detailed information about the survey, as well as the option to edit them is displayed
 - 5.1.2. Changed details are saved on exit

1. Name: View Survey
2. Actor: Coordinator
3. Entry Condition: Pressing the button “View Survey” on the selected survey.
4. Exit Condition: Pressing the button “Return to List Surveys”
5. Flow of Events:
 - 5.1. The coordinator presses the “View Survey” button on the selected survey panel
 - 5.1.1. All the details about the survey is displayed
 - 5.1.2. If the survey is a finished survey
 - 5.1.2.1. The answers to each question is displayed

1. Name: List All Users
2. Actor: Coordinator
3. Entry Condition: Pressing the “List All Users” button
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “List All Users” button
 - 5.2. The list of all active users of the system is displayed
 - 5.2.1. The coordinator can press the information button to display a more detailed information page for the user
 - 5.3. The coordinator can manage the users
 - 5.4. If the user is an Advisor
 - 5.4.1. The “Manage/Select an Advisor” panel is opened for the selected Advisor
 - 5.5. If the user is a Guide
 - 5.5.1. The “Manage/Select a Guide” panel is opened for the selected Guide

1. Name: Search for a User
2. Actor: Coordinator
3. Entry Condition: Pressing the “Search User” button
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “Search User” button
 - 5.2. The coordinator enters the name/email of the user
 - 5.3. If the name/email is incorrect
 - 5.3.1. The necessary error message is displayed

- 5.4. Else the coordinator sees the User and can press “Manage/Select User” or press the information button to display a more detailed information page for the user

1. Name: Remove/Upgrade Guide
2. Actor: Coordinator
3. Entry Condition: Pressing the “Remove/Upgrade Guide” button on the “Manage Guide” page
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “Remove Guide” button
 - 5.1.1. The Guide this page belongs to is removed from the system and their email no longer logs in
 - 5.2. The coordinator presses the “Upgrade Guide” button to upgrade the Guide to Advisor
 - 5.2.1. The email that belongs to the Guide is now updated to Advisor role

1. Name: Remove/Upgrade Candidate Guide
2. Actor: Coordinator
3. Entry Condition: Pressing the “Remove/Upgrade Candidate Guide” button on the “Manage Candidate Guide” page
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “Remove Candidate Guide” button
 - 5.1.1. The Candidate Guide this page belongs to is removed from the system and their email no longer logs in
 - 5.2. The coordinator presses the “Upgrade Candidate Guide” button to upgrade the Candidate Guide to Guide
 - 5.2.1. The email that belongs to the Candidate Guide is now updated to Guide role

1. Name: Remove Advisor
2. Actor: Coordinator
3. Entry Condition: Pressing the “Remove Advisor” button on the “Manage Advisor” page
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “Remove Advisor” button
 - 5.2. The Advisor this page belongs to is removed from the system and their email no longer logs in

1. Name: Manage/Select Advisor
2. Actor: Coordinator
3. Entry Condition: Pressing the “Manage/Select Advisor” button on the “List All Users” page
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:

- 5.1. The coordinator presses the “Manage/Select Advisor” button
- 5.2. A calendar is displayed
- 5.3. The user can choose a day in the week and assigns the advisor to a day

1. Name: Add Guides For Fairs
2. Actor: Coordinator
3. Entry Condition: Pressing the Add Guides For Fairs” button
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “Add Guides For Fairs” button
 - 5.2. The coordinator chooses the fair they want to assign a Guide to
 - 5.3. The coordinator chooses the Guide from the shown Guide list
 - 5.4. The system assigns the given Guide to the given Fair and sends an informatory email including the details of the fair

1. Name: Manage Fair Requests
2. Actor: Coordinator
3. Entry Condition: Pressing the “Manage Fair Requests” button
4. Exit Condition: Pressing “Return to Homepage” button
5. Flow of Events:
 - 5.1. The coordinator presses the “Manage Fair Requests” button
 - 5.2. System shows all pending fair requests with their information
 - 5.3. “Accept” and “Decline” buttons are shown next to the fairs
 - 5.4. If the coordinator presses “Accept”
 - 5.4.1. The Fair is added to the system and is now available for Guide assignment
 - 5.5. Else the coordinator presses “Decline”
 - 5.5.1. The Fair is removed from the pending fair requests

1. Name: See Candidate Guide Information
2. Actor: Candidate Guide
3. Entry Condition: Being logged into the account and is currently viewing the candidate guide panel
4. Exit Condition: Returning to the main page
5. Flow of Events:
 - 5.1 The information is displayed on the candidate guide panel directly

1. Name: See All Available Tours
2. Actor: Candidate Guide
3. Entry Condition: Being logged in and click on the button “Show available tours” on the candidate guide panel
4. Exit Condition: Returning to the main page or returning to the candidate guide panel
5. Flow of Events:
 - 5.1 The candidate guide presses the button that reads “Show available tours”
 - 5.2 If there is no available tours
 - 5.2.1 A textbox that tells the candidate guide that there is no tour that the candidate guide can join is displayed
 - 5.3 If there are joined tours

5.3.1 The list of available tours are displayed with a brief info and all of the list elements displaying a button that reads “Enter”, replacing the candidate guide panel

5.4 Candidate guide can press on the return button

5.4.1 Return button will redirect the candidate guide to the candidate guide panel

1. Name: Register Tour
2. Actor: Candidate Guide
3. Entry Condition: Being logged in and click on a button that reads “Enter” from the list of available tours
4. Exit Condition: Returning to main page or returning to the available tours display menu
5. Flow of Events:
 - 5.1 Every list item in the available tours menu contains a button that reads “Enter”.
 - 5.2 Clicking on one of these buttons directs to a page that shows detailed information about that specific tour, as well as the confirm button
 - 5.3 Candidate guide can click on the confirm button
 - 5.3.1 Confirming the tour adds the candidate guide to that tour
 - 5.3.2 A message is displayed whether the action was successful
 - 5.4 Candidate guide can click on the return button
 - 5.4.1 Return button will redirect the candidate guide to the available tours menu.

1. Name: See Candidate Guide’s Tours List
2. Actor: Candidate Guide
3. Entry Condition: Being logged in and Click on the button “Joined Tours” on the candidate guide panel
4. Exit Condition: Returning to the main page or returning to the candidate guide panel
5. Flow of Events:
 - 5.1 Candidate guide presses the button that reads “Joined Tours”
 - 5.2 If there is no joined tours
 - 5.2.1 A textbox that tells the candidate guide that they have not joined a tour is displayed
 - 5.3 If there are joined tours
 - 5.3.1 The list of tours that the candidate guide has joined are displayed with a brief info and all of the list elements containing a button that reads “Cancel”
 - 5.4 Candidate guide can press on the return button
 - 5.4.1 Return button will redirect the candidate guide to the candidate guide panel

1. Name: Remove Tour
2. Actor: Candidate Guide
3. Entry Condition: Being logged in and click on a button that reads “Cancel” from the list of joined tours
4. Exit Condition: Returning to the main page or returning to the show joined tours menu
5. Flow of Events:

- 5.1 Every list item in the joined tours menu contains a button that reads "Cancel"
- 5.2 Clicking on one of these buttons directs to a page that shows detailed information about that specific tour as well as the confirm button
- 5.3 Candidate guide can click on the confirm button
 - 5.3.1 Confirming the tour removes the candidate guide from that tour
 - 5.3.2 A message is displayed whether the action was successful
- 5.4 Candidate guide can click on the return button
 - 5.4.1 Return button will redirect the candidate guide to the available tours menu.

1. Name: Register Coordinator to System
 2. Actor: Owner
 3. Entry Condition: Being logged in and pressing on the button that reads "Add Coordinator"
 4. Exit Condition: Returning to the main page
 5. Flow of Events:
 - 5.1 The owner presses the button that reads "Register Coordinator"
 - 5.2 The owner enters coordinator information
 - 5.3 The coordinator is added to the system
 - 5.4 A message is displayed accordingly
-
1. Name: List All Coordinators
 2. Actor: Owner
 3. Entry Condition: Being logged in and pressing on the button that reads "Show Coordinators"
 4. Exit Condition: Returning to the main page or returning to the owner panel
 5. Flow of Events:
 - 5.1 The owner presses the button that reads "Show Coordinators"
 - 5.2 If there are not coordinators
 - 5.2.1 A textbox that tells the owner that there is no coordinators in the system
 - 5.3 If there are coordinators
 - 5.3.1 List of all the coordinators are displayed with detailed information and a button on every element that reads "Delete" are displayed
 - 5.4 Owner can click on the return button
 - 5.4.1 Return button will redirect the owner to the owner panel.
-
1. Name: Delete Coordinator
 2. Actor: Owner
 3. Entry Condition: Being logged in and pressing on one of the buttons on the coordinators list item
 4. Exit Condition: Returning to the main page or returning to show coordinators menu
 5. Flow of Events:
 - 5.1 Every list item in the show coordinators menu contains a button that reads "Delete"
 - 5.2 Clicking on one of these buttons shows a pop-up with a confirm button and cancel button, showing the detailed information of the coordinator.
 - 5.3 If owner clicks on the confirm button
 - 5.3.1 Confirming the action removes the coordinator from the system

5.3.2 A message is displayed whether the action was successful

5.4 If owner clicks on the cancel button

5.4 Canceling the action closes the pop-up and shows the list of coordinators

1. Name: List All Data
2. Actor: Coordinator, Owner
3. Entry Condition: Pressing the “List All Data” button on the Data Panel page
4. Exit Condition: Returning to the home page or returning to the Data Panel
5. Flow of Events:
 - 5.1. The user presses the “List All Data” button to view the list of all available data
 - 5.2. The user can press the “Select Data” button to view the data in detail
 - 5.2.1. The user can press the “Edit Data” button to edit the selected data
 - 5.2.2. The user can press the “Remove Data” button to remove the selected data

1. Name: Search Data
2. Actor: Coordinator, Owner
3. Entry Condition: Pressing the “Search Data” button on the Data Panel page
4. Exit Condition: Returning to the home page or returning to the Data Panel
5. Flow of Events:
 - 5.1. The user enters keywords regarding the data they want to search
 - 5.2. If the keywords match
 - 5.2.1. The “Select Data” view of the searched data is shown to the user
 - 5.2.1.1. They can Edit and Remove the selected data
 - 5.2.1.2. They can view detailed information about the selected data
 - 5.3. Else the keywords do not match
 - 5.3.1. Corresponding error message is shown to the user

1. Name: Select Data
2. Actor: Coordinator, Owner
3. Entry Condition: Pressing the “Select Data” button on the listed data or searched data.
4. Exit Condition: Returning to the home page.
5. Flow of Events:
 - 5.1. The user presses the button that reads “Select Data” from next to the data they want to select from the list.
 - 5.1.1. The data is selected
 - 5.2. The user presses the button that reads “Select Data from next to the data they want to select from the search list.”
 - 5.2.1. The data is selected

1. Name: View Data in Detail
2. Actor: Coordinator, Owner
3. Entry Condition: Pressing the button “View Detailed Information” when a data is selected.
4. Exit Condition: Returning to the home page.
5. Flow of Events:

- 5.1. The user presses the button that reads “View Detailed Information” when a data is selected
 - 5.1.1. The detailed information regarding the data’s information is displayed.

1. Name: Edit Data
2. Actor: Coordinator, Owner
3. Entry Condition: Pressing the button “Edit Data” when a data is selected.
4. Exit Condition: Returning to the data panel or to the home page.
5. Flow of Events:
 - 5.1. The user presses the button that reads “Edit Data” when a data is selected
 - 5.1.1. The current data properties is displayed, with ability to change the properties about it
 - 5.1.2. Changed properties are automatically updated once returned to the data panel

1. Name: Apply to join BTO
2. Actor: Bilkent Student
3. Entry Condition: Entering the website with a web browser, then pressing apply to join BTO button
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. After pressing the button, an application form will appear on the screen
 - 5.2. If the student fills all of the application form, send button will appear on the screen
 - 5.3. Student will be notified about whether the process was successful
 - 5.4. Student can also view application form with an user-specific distinction mechanism

1. Name: View Application Form
2. Actor: Bilkent Student
3. Entry Condition: Entering the website, then pressing apply to join BTO button to view the application form.
4. Exit Condition: Pressing the exit button on the page
5. Flow of Events:
 - 5.1. After pressing the button, an application form will be displayed.

1. Name: Edit/Delete Application Form
2. Actor: Bilkent Student
3. Entry Condition: Entering the website with a web browser, then pressing apply to join BTO -> view application form -> edit/delete application form button
4. Exit Condition: Pressing the exit button on the page, deleting the form
5. Flow of Events:
 - 5.1. After pressing the edit/delete button an interface will appear in order to manage the previous application form of the user
 - 5.2. The student can change the contents and delete the form entirely in this page
 - 5.3. If student deletes the form
 - 5.3.1. The form will be deleted permanently, and the student will be notified and directed to the previous screen

1. Name: Login
 2. Actor: BTO Member
 3. Entry Condition: Entering the website with a web browser, then pressing login button
 4. Exit Condition: Pressing the exit button on the page
 5. Flow of Events:
 - 5.1. After pressing the login button two text boxes will appear for username and password, along with forget password button
 - 5.2. Users can enter their information and press login button to enter their profiles
-
1. Name: Forget Password
 2. Actor: BTO Member
 3. Entry Condition: Entering the website with a web browser, then pressing login button and forget password button
 4. Exit Condition: Pressing the exit button on the page
 5. Flow of Events:
 - 5.1. After pressing the forget password button, the user will receive a new password by mail

3. Tech Stack

3.1 ASP.NET

ASP.net core is an open-source, cross-platform backend framework that uses C#. It is fast and efficient, and it offers a secure environment against security breaches and cyber attacks. Moreover; it is easy to use, flexible while also being scalable for large applications. It also has many tools to simplify the developing process such as APIs, plugins, libraries with a built-in package manager called NuGet, garbage collector and a well structured architecture to ease the process of communication [1].

The biggest reason to use ASP.net core is to use the C# language. C# offers an easy to use experience and is highly functional with high efficiency. There are dozens of practical pre-existing libraries and it is full of useful features (async/await, delegate, string interpolations, tuples etc.) and especially considering we will constantly be dealing with lots of data, linq will be a useful feature to easily manipulate the data we have very efficiently. C# programs serve many functionalities for well structured and maintainable code with its getter/setters, nullable types, interfaces and many more. C# is also designed to be easy to read, which is important for a group project.

While ASP.net is on the more niche side on the scale of backend development, there isn't a lack of documentation or resources. It still has active and stable support, a massive community with a large ecosystem and 6% of the websites worldwide are using it.

3.2 React.js

React is a widely used frontend framework that uses Javascript as the programming language. It is used in nearly 40% of web applications hosted today with main examples being Netflix, WhatsApp, Facebook and Twitter. This wide and common use range enables the framework to be flexible for any use case and be greatly documented. Both Javascript

and React are also popular in unofficial work spaces which enhances the available online resources. The documentation, tutorial and online forum count is important in bug fixing and implementation as errors are bound to occur. React also offers JSX where you can define Javascript utilities such as variables as HTML which not only introduces flexibility but also easiness in coding. We have chosen React for our frontend framework because it's easy to learn, well documented and easy to work with nature [2].

3.3 PostgreSQL

PostgreSQL is an open source database with a huge community supporting it. There is a wide range of documentation available on the internet alongside many useful resources.

PostgreSQL prioritizes security and reliability, and it is optional for people who want to manage the data and have full control over it, without worrying about background problems like data loss, conflicts or leaks. So we can focus only on the control over the data. It is also flexible as well as scalable, which is great for future-proofing the project. Alongside this, PostgreSQL supports advanced features for queries, JSON, functions, expressions and constraints. Even with the extensive features, reliability and flexibility, it does not sacrifice its fast performance and efficiency [3].

3.4 Docker

Docker is a useful tool for software developers as it offers a common machine and environment to work on for the developers of the team. This eliminates the dependency, version conflict, operating system and portability problems. It also offers a scalable platform for your application and makes it easy to deploy on the customers machine as you can just deploy the container. As our group utilizes different operating systems and may use different versions of the code, Docker is going to be beneficial for us [4].

3.5 Github

Version control is a critical issue in software development and GitHub is the most common and efficient platform for developers to merge their work without disrupting the workflow of others. It also enables the programmers to see who committed which part of the code when and direct the questions they have or the problems that emerge to the owner of the code. This tech is going to be used in file management and version control in our project in order to prevent data losses and sync up each other's work in a controlled environment [5].

4. References

[1]“ASP.NET CORE: Open-source web framework for .NET,” Microsoft, <https://dotnet.microsoft.com/en-us/apps/aspnet> (accessed Oct. 12, 2024).

[2] “React,” React Blog RSS, <https://react.dev/> (accessed Oct. 12, 2024).

[3] “About,” PostgreSQL, <https://www.postgresql.org/about/> (accessed Oct. 12, 2024).

[4] “Accelerated Container Application Development,” Docker, <https://www.docker.com/> (accessed Oct. 12, 2024).

[5]“About github and git,” GitHub Docs, <https://docs.github.com/en/get-started/start-your-journey/about-github-and-git> (accessed Oct. 12, 2024).