



Bilkent University

Department of Computer Engineering

CS319

Object-Oriented Software Engineering

Classroom Helper

Group 1G

Final Report

Tuna Dalbeler (21802539), Ezgi Saygılı (21802871), Funda Tan (21801861),
Selahattin Cem Öztürk (21802856), Onur Korkmaz(21802925)

Instructor: Eray Tüzün

Teaching Assistants: Elgun Jabrayilzade, Erdem Tuna

Where are we in the production?

	Alpha	
	Groups	
	Review of artifact	
	Instructor View	
	Peer Review	
	Database Connections	←- We are in here
	Beta	Core functionality completed
	User Authorisation	
	Instructor Analytics	
	Better UI	
	Group Work Management Tools	
	Live	
	User Forums	
	Chat	
	Calendar	

We have yet to complete API - front end connections. Unfortunately, for now, front-end only works with hard coded dummy data.

Lessons learnt: Tell us about your experience

Implementation:

We divided the work equally among the group members. We used Visual Studio Code to implement the frontend and Spring boot for the backend. We pushed the code to our Github repository whenever we made a change in the code and we used Zoom for the group meetings.

While implementing the Classroom Helper tool, we tried to comply with the design we promised in the Analysis Report and the Design Report. We somewhat changed the styling we did in the Screen Mock-Ups part of the Analysis Report. This is because we did not have the time to implement the extra features.

We underestimated the time needed for the implementation part. Front-end team did not know any HTML, CSS, Javascript and React. They had to learn all of these languages and libraries to create something functional. On the other hand, the back-end team tried to learn a framework they never used, plus the database. API was not working at the end of the day. Even though someone at the front-end team tried to test their API fetch functions with a fake API, some of them are working, testing for all functions was not possible.

We had no idea how to structure the implementation phase. First thing we did was dividing groups such as the backend and front-end. Apparently that was kind of a wrong approach. Both teams did not really communicate until the end of the deadline. [One of the lessons learned for me (Tuna Dalbeler) was structuring web application development should be more systematic. I will do it again in future, i want to try API first approach.]

Major Changes to the Design:

During back end implementation, we have seperated Forms(feature for reviewing) and Assignments(feature for assigning homeworks)

We excluded calendar, forum and chat functionalities from our project due to time constraint. Also analytic tools for instructors and in-group work management tools are not implemented.

Front end design is not as decent as proposed in screen mock-ups, but the overall structure resembles the screen mock-ups.

During implementation we did not completely stick to the final object design, but made changes wherever necessary. This is because spring boot already had some predefined

functionalities. There are mostly differences in controller classes' implementation as spring boot has a different way of handling requests.

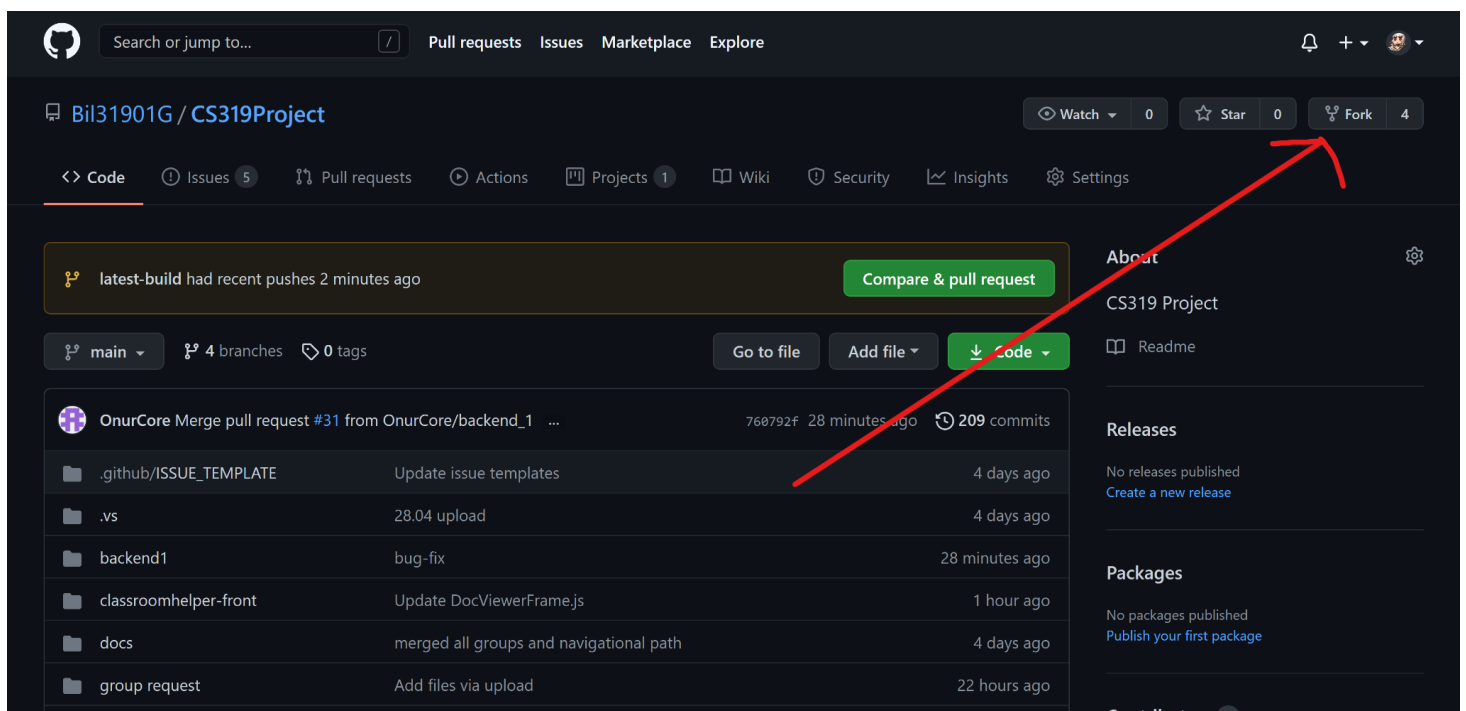
User's Guide

How to Install

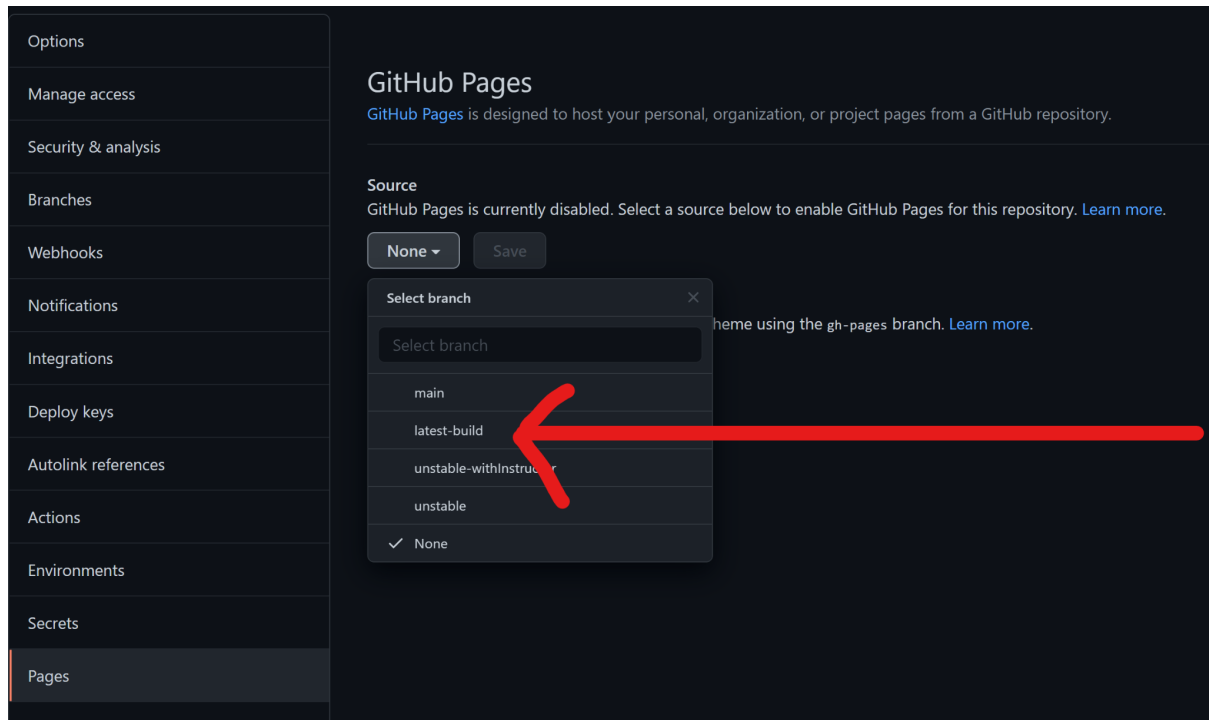
Github link to reach our documents and project:

<https://github.com/Bil31901G/CS319Project>

- A) Deploying front-end
- 1) Go to the github link above.
 - 2) Fork the repository



- 3) Go to settings/pages.
- 4) Select latest-build



- 5) Click save
- 6) Site is published at {username}.github.io/CS319Project/
It might take a few minutes to actually publish the site.
- 7) If wanted a custom domain can be selected from settings/pages.

An example from an old build.

<https://contactlost.github.io/>

If publishing on github pages is not wanted. Download the latest-build and deploy it to another server.

- B) Deploying backend
- 1) Choose your favorite machine.
 - 2) Install jar file from this link.

<https://github.com/Bil31901G/CS319Project/releases/tag/v0.1>

- 3) Install MySQL
- 4) Run the jar file.
- 5) If needed, give necessary firewall permissions.

Work Allocation

Onur Korkmaz:

- Analysis Report
 - User Interface and Navigational Paths
 - Screen Mock-ups
- Design Report
 - Subsystem Decomposition
 - Low level design
- Back end
 - Implementation of classes in object model, connecting to database and testing

Tuna Dalbeler:

- Analysis Report
 - Sequence Diagrams
- Design Report
 - Low level design
- Front end
 - All groups system
 - Artifact review system
 - Navigation system
 - Peer Evaluation system
 - Assignment Upload system
 - Made all the subsystems for front end
 - Made a fake API to test web-side API functions.
 - Made the demo video

Emine Ezgi Saygılı:

- Analysis Report
 - Object and class model
- Design Report
 - Low level design
- Back end
 - Implementation of classes in object model, connecting to database and testing

Funda Tan:

- Analysis Report
 - Activity Diagrams
 - State Diagram
- Design Report
 - System Architecture
- Front end
 - Login and sign-up pages

- Css files for styling
- Router system for the pages in the Navigation Bar
- My Group page (Group members list, Assignments list, Join requests list)

Selahattin Cem Öztürk:

- Analysis Report
 - Use Case Diagram.
 - Explanation of the use cases
 - State diagram
 - Half of the mock-ups
- Design Report
 - Low level design.
 - Access control matrix
- Front end
 - Entrance page
 - Instructor's login and signup page
 - Instructor's assignment page
 - Assignment upload page
 - People page and people list
 - The page that show all the courses in the system
 - Add course page
 - File dialogue button
 - Router system for the pages in the Navigation Bar for Instructor