



CS 353 Database Systems

Project Final Report

Coding Practice and Interview System

MasterCoding

Group 16

Ali Kemal Özkan | 21302087 - Sec 3

Osman Can Yıldız | 21302616 - Sec 3

Göktuğ Öztürkcan | 21702290 - Sec 2

Tanay Toksoy | 21703919 - Sec 2

22th May, 2020

Project Website

<https://alikelmalozkan.github.io/MasterCoding/>

Table of Contents

Description of The System	2
E-R Diagram	3
Table Schemas	4
3.1. User	4
3.2. Admin	4
3.3. Developer	4
3.4. Company Representative	4
3.5. Company	4
3.5. Question	5
3.6. Interview	5
3.7. Interview Question	5
3.8. Track	5
3.9. Test case	5
3.10. Category	5
3.11. Submission	5
3.12. Admin_Question	6
3.13. Question_Track	6
3.14. Interview_Interview_Question	
Relational Model:	6
3.15. User_Track	6
3.16. Developer_Submission	6
Implementation Details	6
User Manual	8
5.1. Sign Up	8
5.2. Login	9
5.3. Dashboard	10
5.4. Question List	12
5.5. Ask Question Page	13
5.6. Solve Question Page	14
5.7. Leaderboard	15
5.8. Choose Track	16
5.9. Track Progress	17
5.10. Submit Track Question	18
5.11. Job Interview List	19
5.12. Job Interview Question List	20
5.13. Company Interview Requests	21
5.14. Prepare Track Page	21
5.15. Interview Track List for Company	22
5.16. Profile Page for Users	23

1. Description of The System

In this project, we made a Coding Practice Database System -which is called mastercoding- web-based application built to monitor and maintain the standing of various tracks that include coding questions for both developers and job interviewees. Moreover, the system monitors leaderboard tables with points ,from high to low, according to specific tracks. The system feature give information about user, in this case developer, interviewees and company representatives and also it provides coding questions with the purpose of practice or a job interview.

Users who are developers and company representatives sign up the system according to their status. To clarify, if a developer will sign up, he/she sign-ups from the developer sign up button and company representative sign-ups from the company representative sign up button. Then they are directed to log-in page. Users can login to the system with their e-mail and password. After having logged-in, users are directed to dashboard page which includes buttons for tracks, leaderboard, user questions.

Developers can see questions, ask questions, join track, see leaderboard and join interviews. Admins can approve job interview requests and question requests. Companies can see leaderboard, can arrange job interviews. They can prepare interview tracks for developers. They can ask specific questions.

With this system we aim to put together developers and companies. In addition to that, all developer can benefit from each other with Mastercoding.

This report contain final structure of our project. Final E-R, table schemas, user's manual and implementation details are included in this report.

2. E-R Diagram

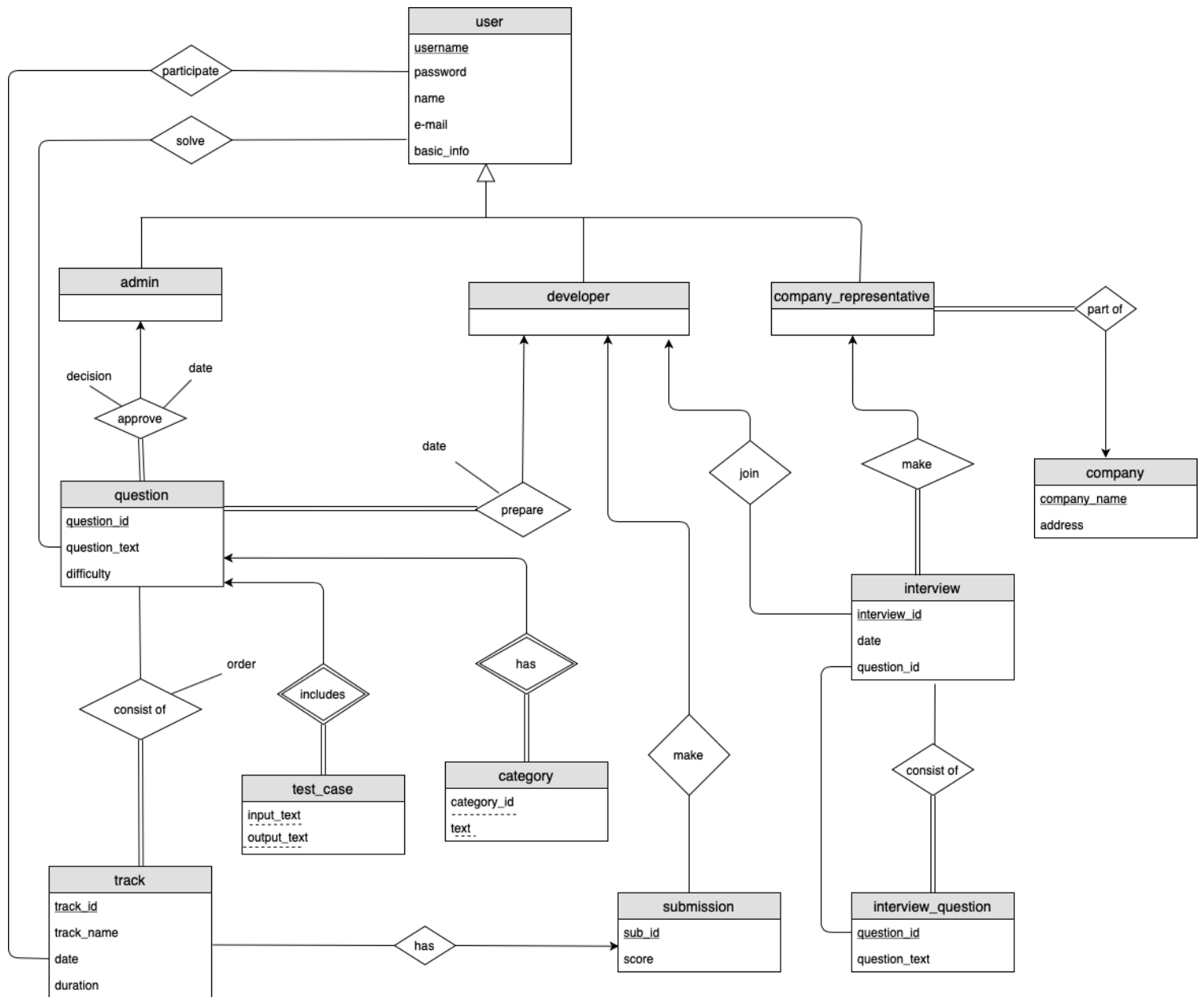


Figure 1: E-R Diagram

3. Table Schemas

3.1. User

Relational Model:

user(username, password, name, e-mail, basic_info)

Functional Dependencies:

username \rightarrow password, name, e-mail, basic_info

3.2. Admin

Relational Model:

admin(username)

3.3. Developer

Relational Model:

developer(username)

3.4. Company Representative

Relational Model:

company_representative(username, company_name)

FOREIGN KEY (username) references (user),

FOREIGN KEY (company_name) references (company)

);

3.5. Company

Relational Model:

company(company_name, address, username)

Functional Dependencies:

company_name \rightarrow address

3.5. Question

Relational Model:

question(question_id, username, question_text, difficulty)

Functional Dependencies:

question_id \rightarrow question_text, difficulty

3.6. Interview

Relational Model:

interview(interview_id, date, question_id, username)

Functional Dependencies:

interview_id \rightarrow date, question_id

3.7. Interview Question

Relational Model:

interview_question(question_id, question_text)

Functional Dependencies:

question_id \rightarrow question_text

3.8. Track

Relational Model:

track(track_id, track_name, date, duration)

Functional Dependencies:

track_id \rightarrow track_name, date, duration

3.9. Test case

Relational Model:

test_case(question_id, input_text, output_text)

Functional Dependencies:

3.10. Category

Relational Model:

category(question_id, category_id, text)

3.11. Submission

Relational Model:

submission(sub_id, score)

Functional Dependencies:

sub_id \rightarrow score

3.12. Admin_Question

Relational Model:

admin_question(username, question_id, date, decision)

3.13. Question_Track

Relational Model:

question_track(question_id, track_id, order)

3.14. Interview_Interview_Question

Relational Model:

interview_interview_question(interview_id, question_id);

3.15. User_Track

Relational Model: user_track(username, track_id, track_name, date, duration);

Functional Dependencies:

User_Question

3.16. Developer_Submission

Relational Model: developer_submission(username, sub_id, score);

4. Implementation Details

In the implementation, we used many programmes. For frontend of our project, we used Photoshop CC to design UI. Then by using HTML, we created a website with our own design. For backend, MySQL is used to create a database server. To connect backend server and frontend together, we used Spring Boot Framework which is written by Java programming language and which includes SQLs and Spring Boot Libraries. Since Spring Boot is a connector for backend and frontend, Spring Boot has connected to our UI by Javascript in frontend and also connected to database server which is a local host that works at <http://localhost:3306>. Spring Boot has also a local host at <http://localhost:8080>. To test HTTP Requests which are POST, PUT, GET and DEL, we used POSTMAN before creating UI. The tests that are used in POSTMAN can be tested as

<http://localhost:8080/Company/get{3}>} for instance. In this example, we get the information of a company whose id is 3 from our database.

Contribution of Work

Osman Can Yıldız

Backend:

- Spring Boot with Java
- POSTMAN test
- MySQL

Frontend:

- Photoshop UI Design
- HTML
- Javascript

Ali Kemal Özkan

Backend:

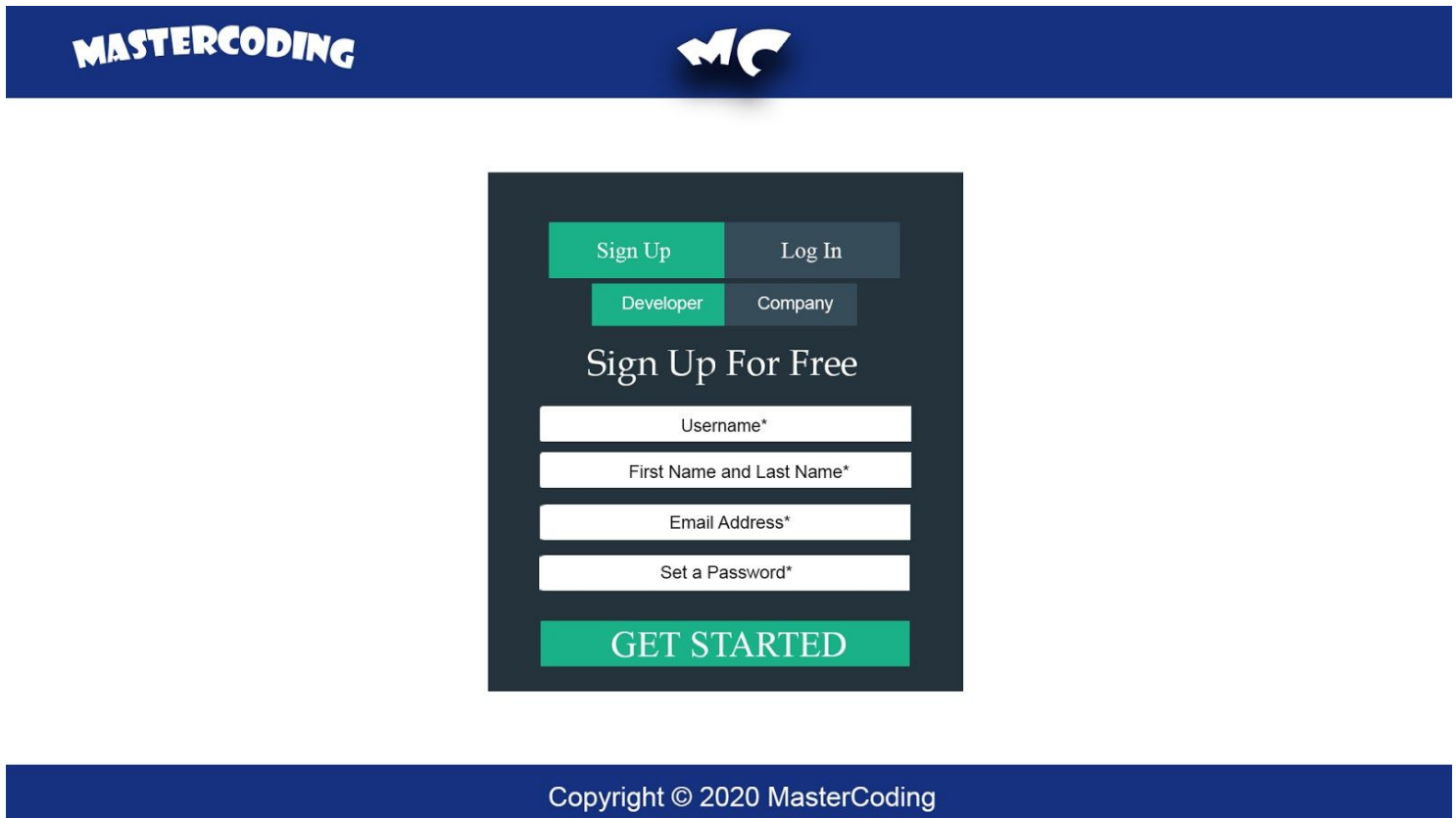
- POSTMAN test
- MySQL
- SQL

Frontend:

- HTML and page transitions
- Javascript

5. User Manual

5.1. Sign Up

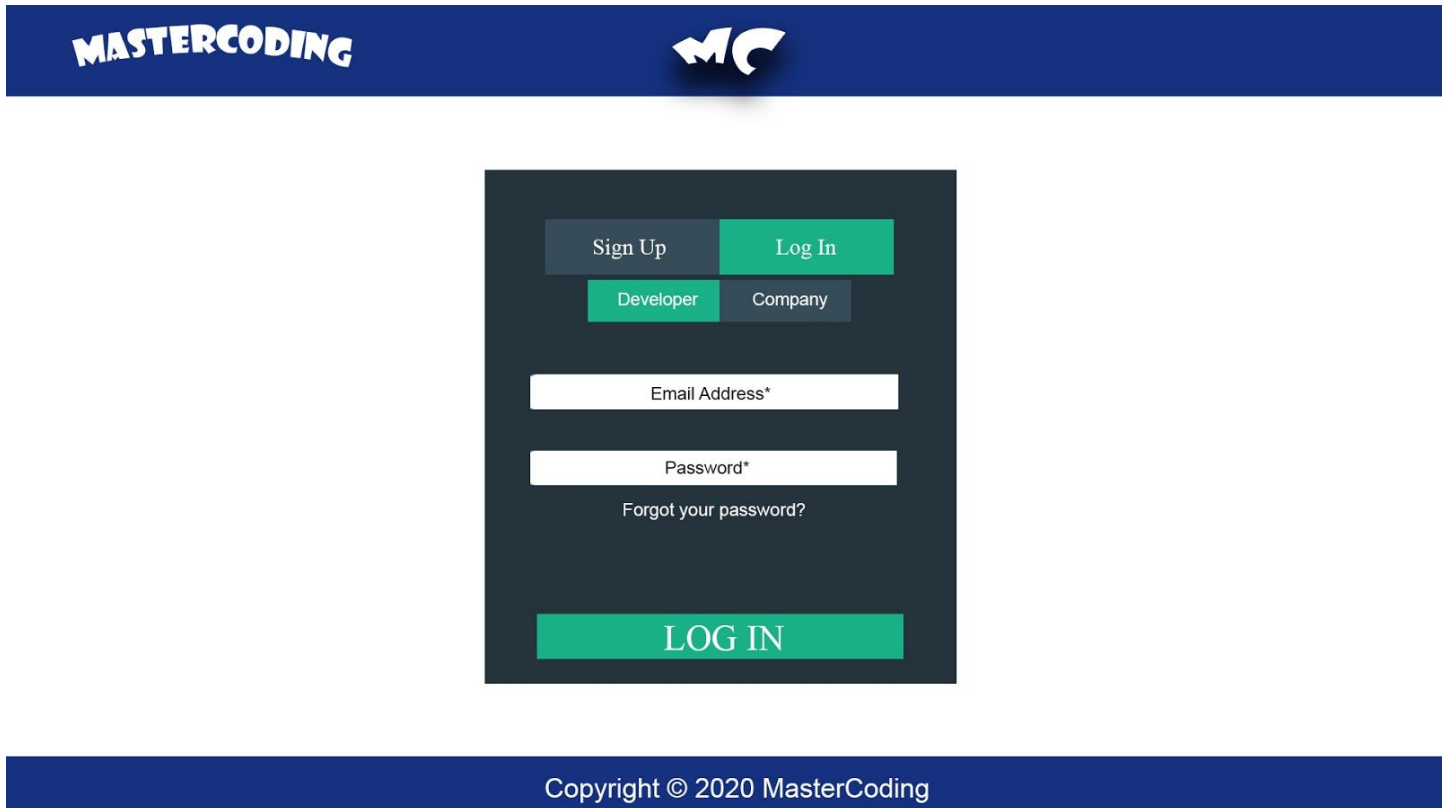


The image shows a web application interface for MasterCoding. At the top, there is a dark blue header bar with the 'MASTERCODING' logo on the left and a stylized 'MC' logo on the right. Below the header, the main content area is a dark grey rectangle. Inside this rectangle, there are two buttons at the top: 'Sign Up' (green) and 'Log In' (grey). Below these are two more buttons: 'Developer' (green) and 'Company' (grey). The text 'Sign Up For Free' is centered below the buttons. Underneath, there are four input fields: 'Username*', 'First Name and Last Name*', 'Email Address*', and 'Set a Password*'. At the bottom of the form is a large green button labeled 'GET STARTED'. At the very bottom of the page, there is a dark blue footer bar with the text 'Copyright © 2020 MasterCoding'.

Figure 2: Sign up screen

Firstly, users should sign up with username, full name, e-mail and password as a developer or company representative. After filling the sign up form, they can click on get started button and login.

5.2. Login



The image shows the MasterCoding login interface. At the top is a dark blue header with the 'MASTERCODING' logo on the left and a stylized 'MC' logo on the right. Below the header is a dark grey login form. The form contains two buttons at the top: 'Sign Up' (grey) and 'Log In' (green). Below these are two more buttons: 'Developer' (green) and 'Company' (grey). The 'Developer' button is selected. Below the role buttons are two input fields: 'Email Address*' and 'Password*'. Below the password field is a link that says 'Forgot your password?'. At the bottom of the form is a large green button labeled 'LOG IN'. At the very bottom of the page is a dark blue footer with the text 'Copyright © 2020 MasterCoding'.

Figure 3: Login screen for developers

Users and company representatives can login with their e-mail address and password. After login, by clicking mastercoding on left top, users can go back to dashboard.

5.3. Dashboard

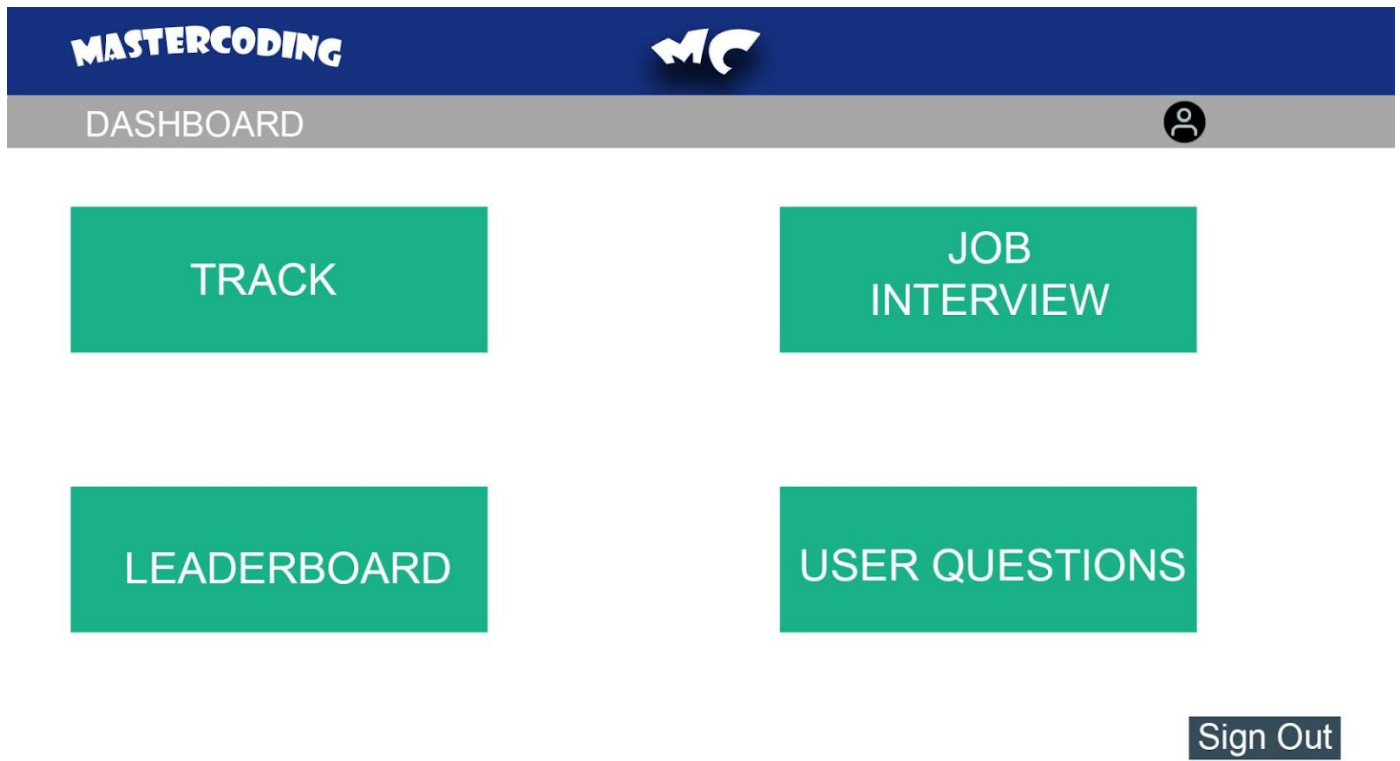


Figure 4: Dashboard for Developer

Dashboard screen works like a main menu where the developer can either access a track, job interview, leaderboard or the user questions. After login developer can see job interview, questions, leaderboard and tracks.

Admins can see job interview requests, and developer questions requests within their dashboard.

Companies can look at job interview menu and leaderboard. Companies can prepare interviews for developers.

All user can sign out with using sign out button in dashboard.



Figure 5: Dashboard for Admin

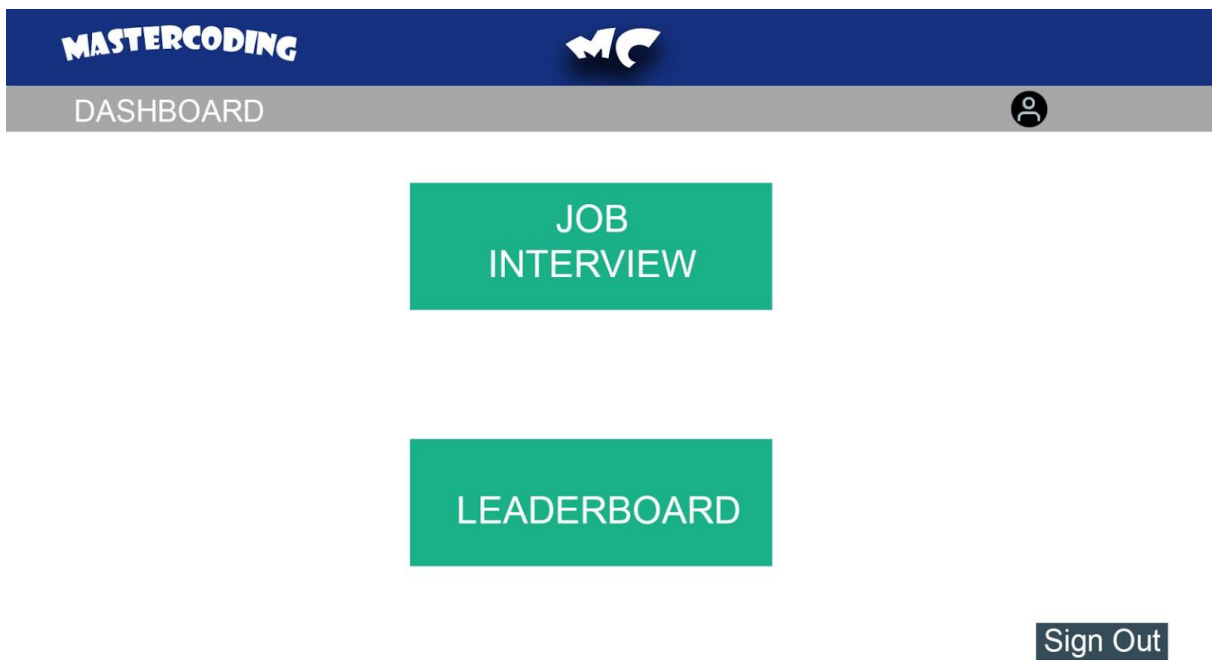


Figure 6: Dashboard for Company

5.4. Question List

MASTERCODING

MC

USER QUESTIONS

Developer13

User Name	Question	Category	Level
developer13	How do I convert a String to an int i...	String	Easy
developer12	How to convert List<Integer> to int[...	Loop	Medium
developer1	How to convert List<Integer> to int[...	Loop	Hard
developer3	How to convert List<Integer> to int[...	Loop	Easy
developer23	How do I convert a String to an int i...	Linked List	Medium
developer33	How to convert List<Integer> to int[...	Funtion	Hard
developer43	How to convert List<Integer> to int[...	Method	Hard
developer53	How to convert List<Integer> to int[...	String	Hard
developer63	How to convert List<Integer> to int[...	String	Easy
developer73	How to convert List<Integer> to int[...	String	Easy

To see or answer the question, press the question.

Ask Question

DASHBOARD

Figure 7: Question List

This screen shows the question list created by the developers which are all categorized and separated by their difficulty level. Ask question button allows the users to create their own questions. Dashboard button takes the user back to Dashboard. Every developer can ask questions to other users. Every user can see these question after getting approval from admin.

5.5. Ask Question Page

MASTERCODING MC

USER QUESTIONS Developer13

Your Question

Given a string, return a new string made of 3 copies of the last 2 chars of the original string. The string length will be at least 2.

TEST CASE

extraEnd("Hello") → "lololo"
extraEnd("ab") → "ababab"
extraEnd("Hi") → "HiHiHi"

Category

String
Loop
Funtion
Method
...

Date : 09.04.2020

Difficulty

Easy Medium Hard

SUBMIT

User Questions

Figure 8: Ask Question Screen for Developer

By clicking ask question button developers can submit a question with test cases. Developer have to choose difficulty and category of the question. Also companies can prepare question for interviews from their job interview menu.

MASTERCODING MC

INTERVIEW QUESTION Nokia

Question 1

Given a string, return a new string made of 3 copies of the last 2 chars of the original string. The string length will be at least 2.

TEST CASE

extraEnd("Hello") → "lololo"
extraEnd("ab") → "ababab"
extraEnd("Hi") → "HiHiHi"

Category

String
Loop
Funtion
Method
...

Difficulty


Easy Medium Hard

CONFIRM

TRACK LIST

Figure 9: Prepare Question Page for Company

5.6. Solve Question Page

MASTERCODING

USER QUESTIONSDeveloper13

QUESTION

Given a string, return a new string made of 3 copies of the last 2 chars of the original string. The string length will be at least 2.

TEST CASE

```
extraEnd("Hello") → "lololo"  
extraEnd("ab") → "ababab"  
extraEnd("Hi") → "HiHiHi"
```

Your Answer

COMPILE

ASKED BY

Developer12

Date : 09.04.2020

2,5 B

33

User Questions

Figure 10: Solve Question Page for Developer Questions

In this screen the user can see the questions asked by other developers and then compile their programs with a test case and then submit their answers. If the User Questions is pressed it, lists all the user questions again. Users can also like or dislike the question asked.

5.7. Leaderboard

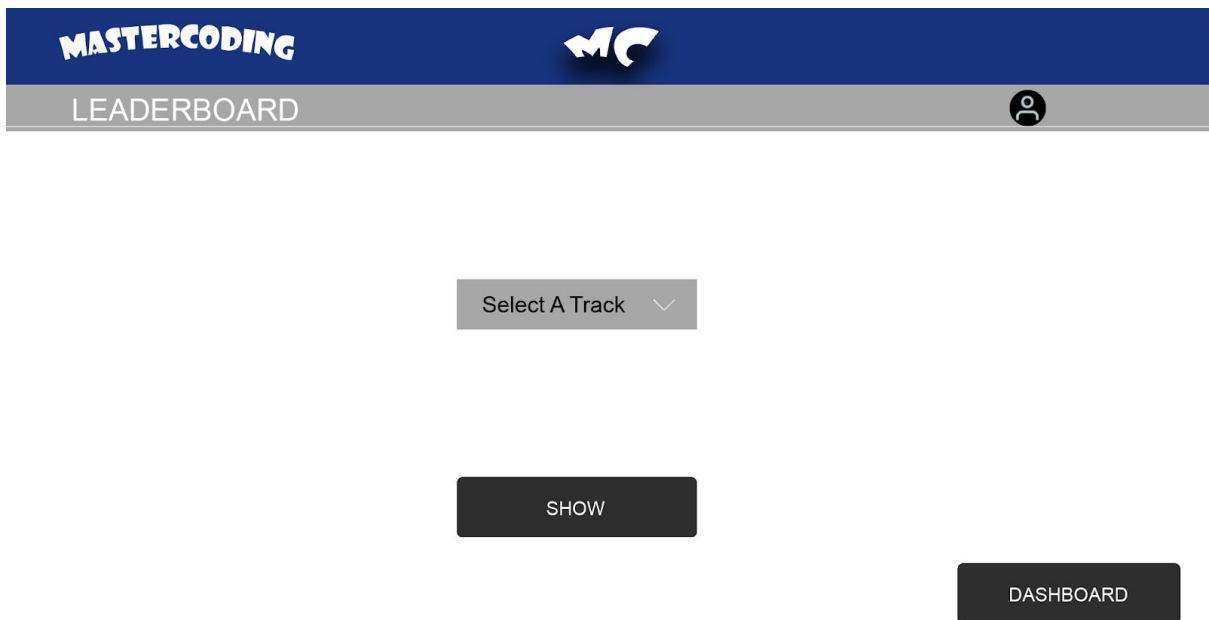


Figure 11: Leaderboard Track Choose Page

We have different leaderboards for different tracks. Developers or companies should choose a track first.

The screenshot shows the top navigation bar with the 'MASTERCODING' logo on the left and the 'MC' logo on the right. Below the navigation bar is a grey bar with the word 'LEADERBOARD' on the left and a user profile icon labeled 'Developer13' with a dropdown arrow on the right. Below the navigation bar, there is a table with 10 rows of user data. To the right of this table is a smaller table titled 'YOUR PLACE' showing the user's current position. Below the 'YOUR PLACE' table is a dark grey button labeled 'DASHBOARD'.

	USER	POINT
1.	Developer13	990
2.	Developer15	980
3.	Developer17	970
4.	Developer14	960
5.	Developer1	950
6.	Developer2	940
7.	Developer23	930
8.	Developer33	920
9.	Developer63	910
10.	Developer93	900

YOUR PLACE

	USER	POINT
1.	Developer13	990

Figure 12: Leaderboard

Leaderboard screen shows the leaderboard and specifically shows the current place of the user in the leaderboard. Companies also can see that without your place part.

5.8. Choose Track

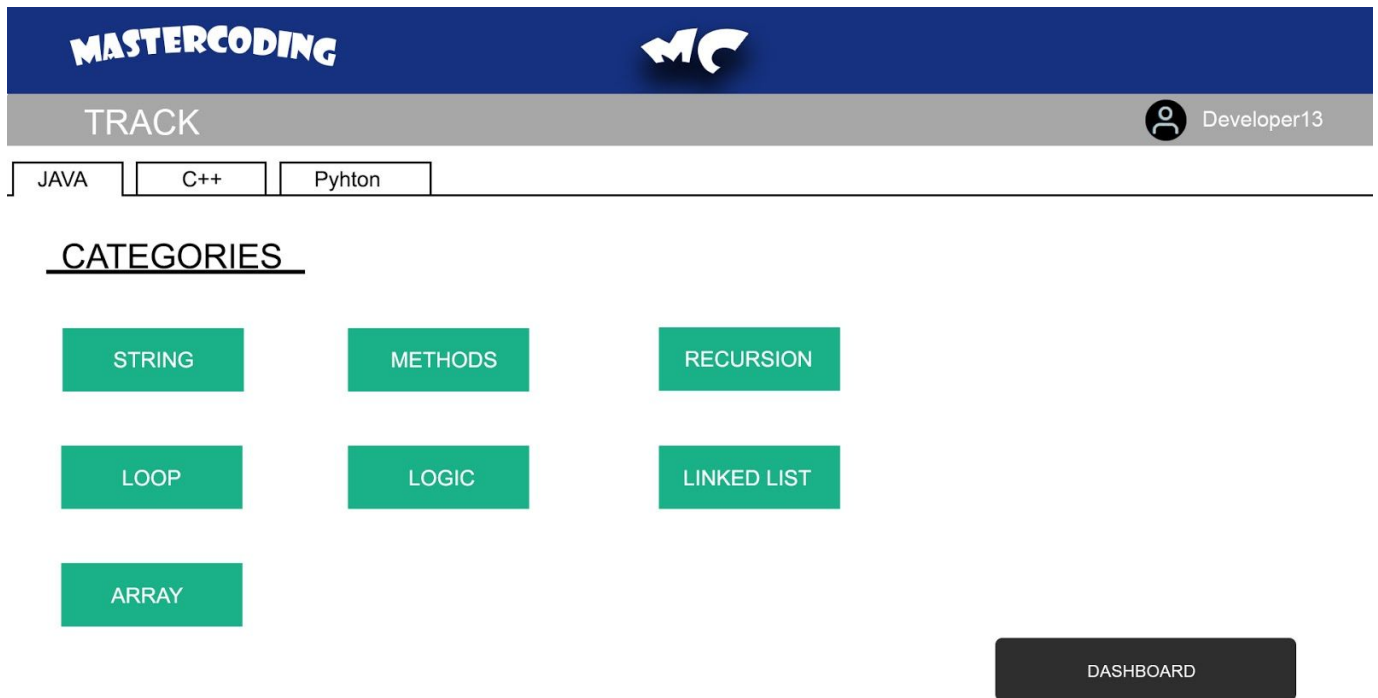


Figure 13: Choose Track Page

This acts as the main screen for tracks from where users can choose the programming language and then choose a category for which he/she will take part in. Or the user can return to the dashboard. User can choose languages and categories for a track to solve.

5.9. Track Progress

MASTERCODING

MC

TRACK

Developer13

JAVA

C++

Pyhton

TRACK

STRING

EASY

- Question 1 (DONE)

- Question 2

- Question 3

- Question 4

- Question 5

- Question 6

- Question 7

- Question 8

- Question 9

- Question 10

- Question 11

- Question 12

MEDIUM

- Question 1

- Question 2

- Question 3

- Question 4

- Question 5

- Question 6

- Question 7

- Question 8

- Question 9

- Question 10

- Question 11

- Question 12

HARD

- Question 1

- Question 2

- Question 3

- Question 4

- Question 5

- Question 6

- Question 7

- Question 8

- Question 9

- Question 10

- Question 11

- Question 12

NOTICE: After selected the level,
the time for questions starts!..

Time Left: 59 minutes

Date: 03.04.2020


FINISH TRACK

DASHBOARD

Figure 14: Track Progress Page

Tracks main screen shows which questions have been completed and time remaining. Through here users can access their next questions or return to the dashboard. Users can finish the track with clicking on finish track button. They can see their progress.

5.10. Submit Track Question

MASTERCODING

TRACKDeveloper13

STRING - QUESTION 1

QUESTION

Given a string, return a new string made of 3 copies of the last 2 chars of the original string. The string length will be at least 2.

TEST CASE

extraEnd("Hello") → "lololo"
extraEnd("ab") → "ababab"
extraEnd("Hi") → "HiHiHi"

CODE BLOCK







```
public String extraEnd(String str) {  
    String temp="";  
    for(int i = 0; i<3 ; i++)  
        temp += str.substring(str.length()-2,str.length());  
    return temp;  
}
```


COMPILESUBMIT

Expected

extraEnd("Hello") → "lololo"
extraEnd("ab") → "ababab"
extraEnd("Hi") → "HiHiHi"
extraEnd("Candy") → "dydydy"
extraEnd("Code") → "dedede"
other tests

Run

"lololo"	OK	
"ababab"	OK	
"HiHiHi"	OK	
"dydydy"	OK	
"dedede"	OK	
	OK	

 All Correct

PRACTICE LIST

Time Left: 59 min

Figure 15: Submit Track Question Page (Correct answer)

Question screen on track shows the question and the user can enter the answer and compile his/her code and see results from the test cases. After user submits the answer and it is correct then the user proceeds on to the next question automatically. Users can see time left, test cases and run infos from that page.

MASTERCODING

MC

TRACK

Developer13

STRING - QUESTION 1

QUESTION

Given a string, return a new string made of 3 copies of the last 2 chars of the original string. The string length will be at least 2.

TEST CASE

```
extraEnd("Hello") → "lololo"
extraEnd("ab") → "ababab"
extraEnd("Hi") → "HiHiHi"
```

CODE BLOCK

```
public String extraEnd(String str) {

    return "hello";

}
```

COMPILER

SUBMIT

Expected

extraEnd("Hello")	→	"lololo"
extraEnd("ab")	→	"ababab"
extraEnd("Hi")	→	"HiHiHi"
extraEnd("Candy")	→	"dydydy"
extraEnd("Code")	→	"dedede"
other tests		

Run

"hello"	X	
"hello"	X	
"hello"	X	
"hello"	X	
"hello"	X	
	X	

TRY AGAIN!

Practice List

Time Left: 59 min

Figure 16: Submit Track Question Page (False answer)

If the user code is wrong after compilation the test cases will be checked and will be shown to the user which test cases worked and which did not.

5.11. Job Interview List

MASTERCODING

MC

JOB INTERVIEW

Developer13

Company List

Name	Interview Question
Nokia	Question link
Bosch	Question link
Vestel	Question link
Huawei	Question link
Vatan Computer	Question link
Yemek Sepeti	Question link
Riot Games	Question link
Blizzard Entertainment	Question link
Valve	Question link
Ubisoft	Question link

DASHBOARD

Figure 17: Interviews List

This page shows all the interviews available to the user. User can take the interviews by clicking on them or either return to dashboard by clicking the dashboard menu. User can join an interview within this list.

5.12. Job Interview Question List

The screenshot shows the MasterCoding Interview interface. At the top, there is a dark blue header with the 'MASTERCODING' logo on the left and a stylized 'MC' logo on the right. Below the header, a grey bar contains the word 'INTERVIEW' on the left and a user profile icon with the text 'Developer13' on the right. The main content area is white. On the left, under the heading 'NOKIA COMPANY', there is a list of 12 questions. The first question, 'Question 1 (DONE)', is marked as completed. The remaining questions are 'Question 2' through 'Question 12'. On the right, there is a notice: 'NOTICE: After selected the level, the time for questions starts!..'. Below the notice, a green-bordered box displays 'Time Left: 60 minutes' and 'Date: 03.04.2020'. At the bottom right, there are two buttons: 'FINISH INTERVIEW' and 'DASHBOARD'.

MASTERCODING **MC**

INTERVIEW Developer13

NOKIA COMPANY

- Question 1 (DONE)
- Question 2
- Question 3
- Question 4
- Question 5
- Question 6
- Question 7
- Question 8
- Question 9
- Question 10
- Question 11
- Question 12

NOTICE: After selected the level, the time for questions starts!..

Time Left: 60 minutes
Date: 03.04.2020

FINISH INTERVIEW

DASHBOARD

Figure 18: Interviews List

This screen shows the process of the user in interview user has a time limit and can see their whole process through this screen. By clicking finish interview user can end the interview or the user can return to dashboard by clicking on dashboard button.

5.13. Company Interview Requests

MASTERCODING

MC

COMPANY INTERVIEW REQUESTS

Admin_1

Company	Interview Track Name	
Nokia	Intern	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Riot Games	Software Engineer	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>
Intertech	Project Manager	<input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>

☒ Confirm
☐ Deny
☐ Inspect

DASHBOARD

Figure 19: Company Interview Requests

In this page admin can look at interview requests and make confirm or deny. Companies can send interview requests to specific user.

5.14. Prepare Track Page

MASTERCODING

MC

TRACK

Nokia

TRACK LIST

Track Name: Internship

- Question 1

- Question 2

- Question 3

- Question 4

- Question 5

- Question 6

- Question 7

- Question 8

- Question 9

- Question 10

- Question 11

- Question 12

Add Question

Remove Question

Edit Question

Programming Language

1. JAVA

2. C++

3. PYTHON

Total Time: mins

Date:/../..

RELEASE TRACK

DASHBOARD

Figure 20: Prepare Track Page

This page is preparing interview track for companies. Companies can prepare interviews as they want to. Companies can determine language, time, date etc. They can add or remove question as they want.

5.15. Interview Track List for Company

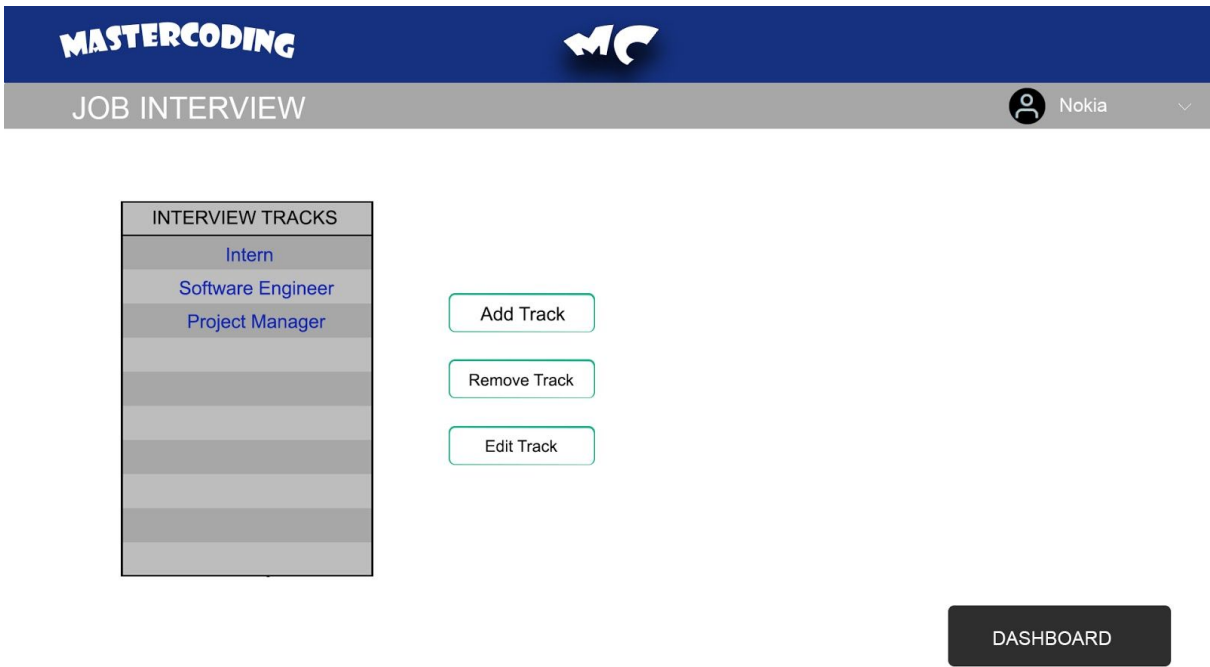


Figure 21: View Interview List

This page is viewing interview track list for companies. Companies can arrange that list. They can add-remove and edit a track.

5.16. Profile Page for Users

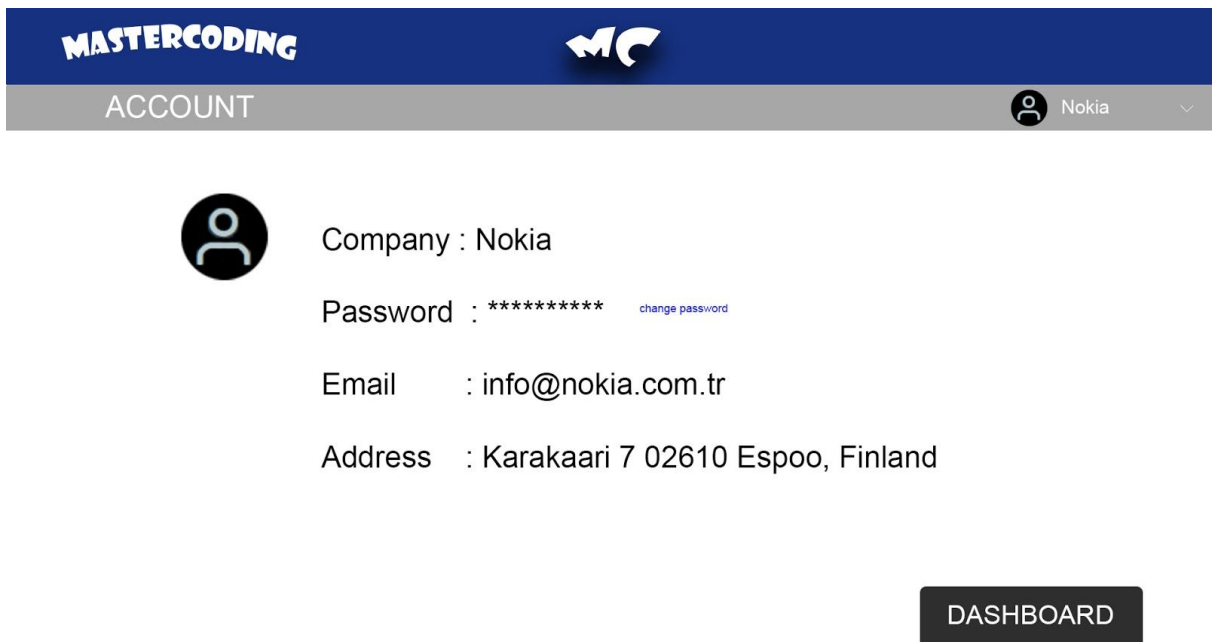


Figure 22: Company Profile Page

Users can see their profile page information. By clicking their name on right top, they can see their profiles. Companies have, company name, password, e-mail and address info.

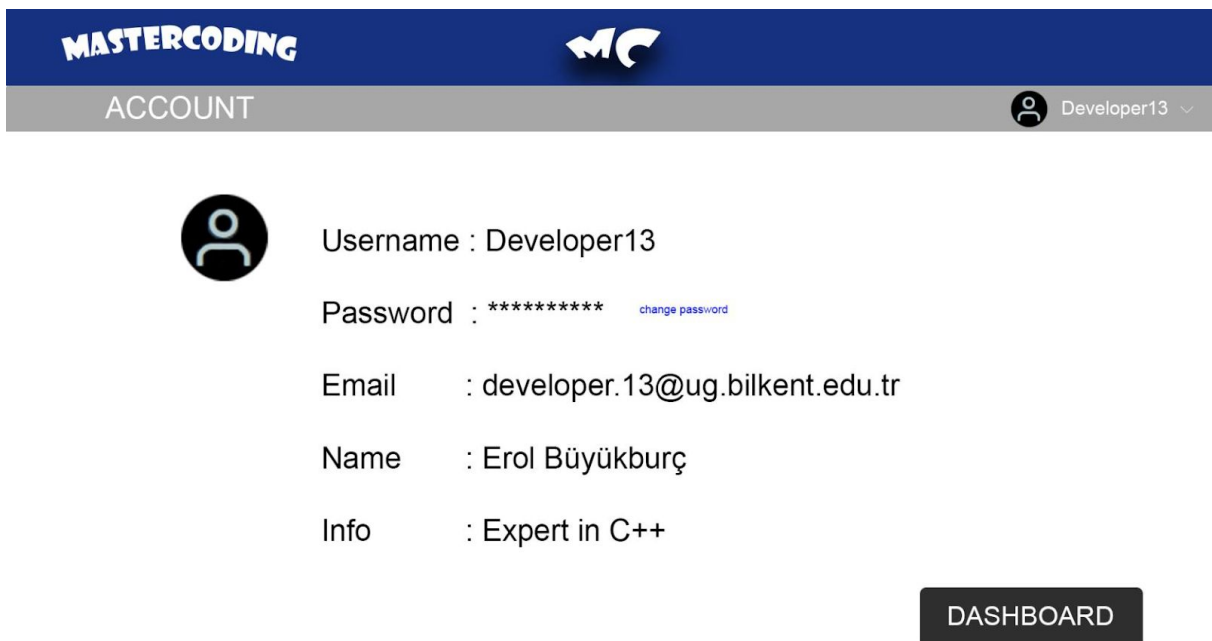


Figure 23: Developer Profile Page

Users can see their profile page from the right top of the corner. They can change password from here.