

Blazor Web Documentation

BR20-2

2021

Foreword

Thank you for granting me the opportunity to be a candidate to join the Research and Development team in Software Laboratory Center at Binus University. Thank you for everyone at Operation Division for leading me throughout the semester, everyone at Research and Development for choosing me to be a candidate, and everyone at Software Laboratory Center who has helped and accompanied me to reach the state I am currently at, especially my generation (20-2). I am grateful and will try my best at doing this test.

Technologies

ASP.NET Core 5.0

ASP.NET Core 5.0 is the latest web framework for developing web applications on the .NET platform by Microsoft. It is better from its predecessor, ASP.NET 4.x, as ASP.NET Core is much more performant, provides cross platform solution, and its latest UI technology, Blazor.

ASP.NET Core Identity

ASP.NET Core Identity provides authentication and authorization scaffolding to ASP.NET Core projects with built-in supports for login register, social login, database migration, and more.

Blazor (Server)

Blazor is part of ASP.NET to build interactive web UI with C#. There are two hosting models for Blazor, Blazor WebAssembly and Blazor Server. Blazor Server is used in this application to lighten up client's load and prerender pages server-side before sending them to client.

Entity Framework Core + SQL Server

Entity Framework Core is used as ORM in this application together with SQL Server to help in writing data-access code and optimizing database queries.

MudBlazor

MudBlazor is material themed components made for blazor.

Bootstrap 4.3.1

ASP.NET Core project template comes with Bootstrap 4.3.1 out of the box to enable rapid styling.

Pages

All pages are placed under “Pages” folder.

Home

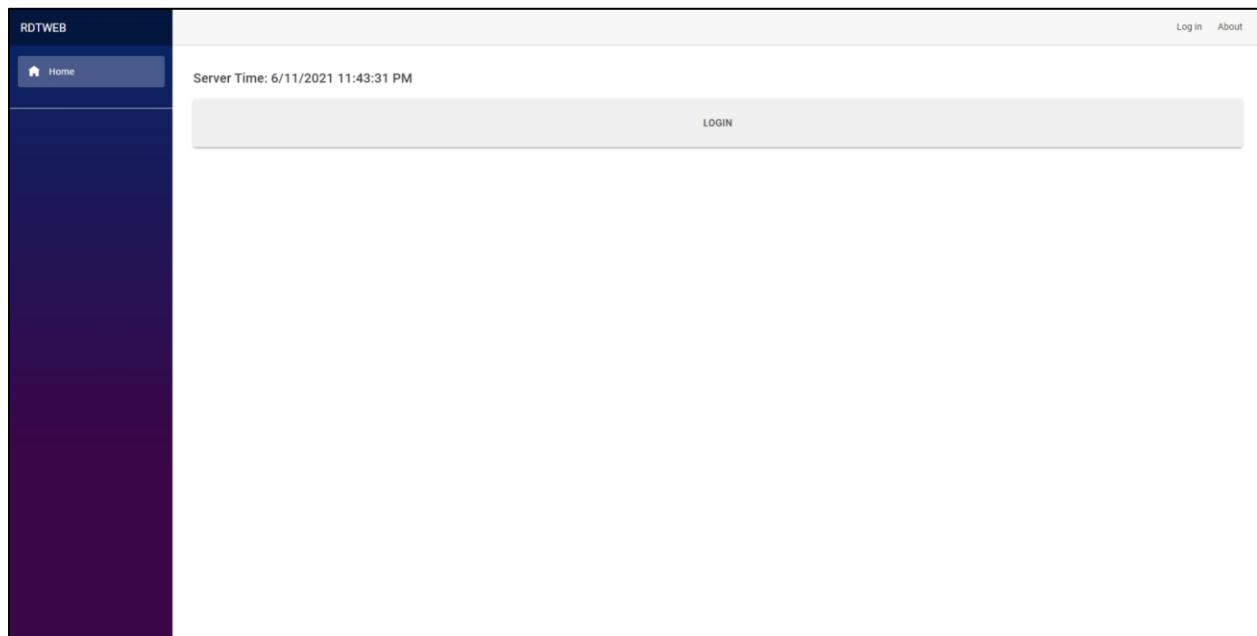


Figure 1. Home (guest).

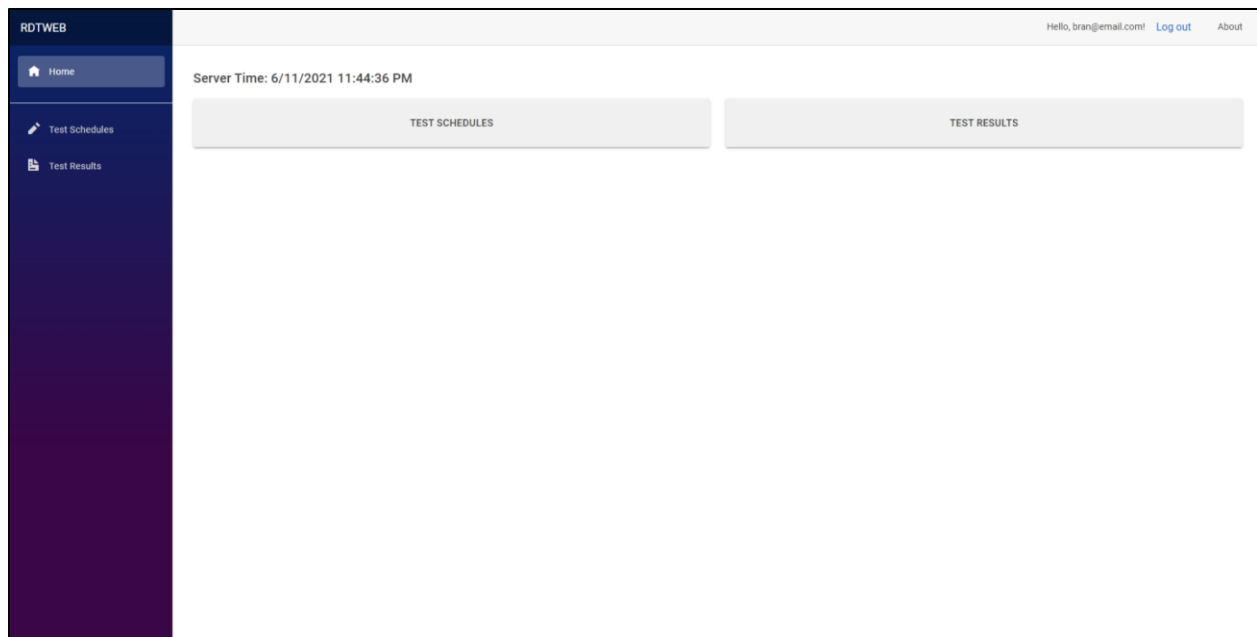


Figure 2. Home (participant).

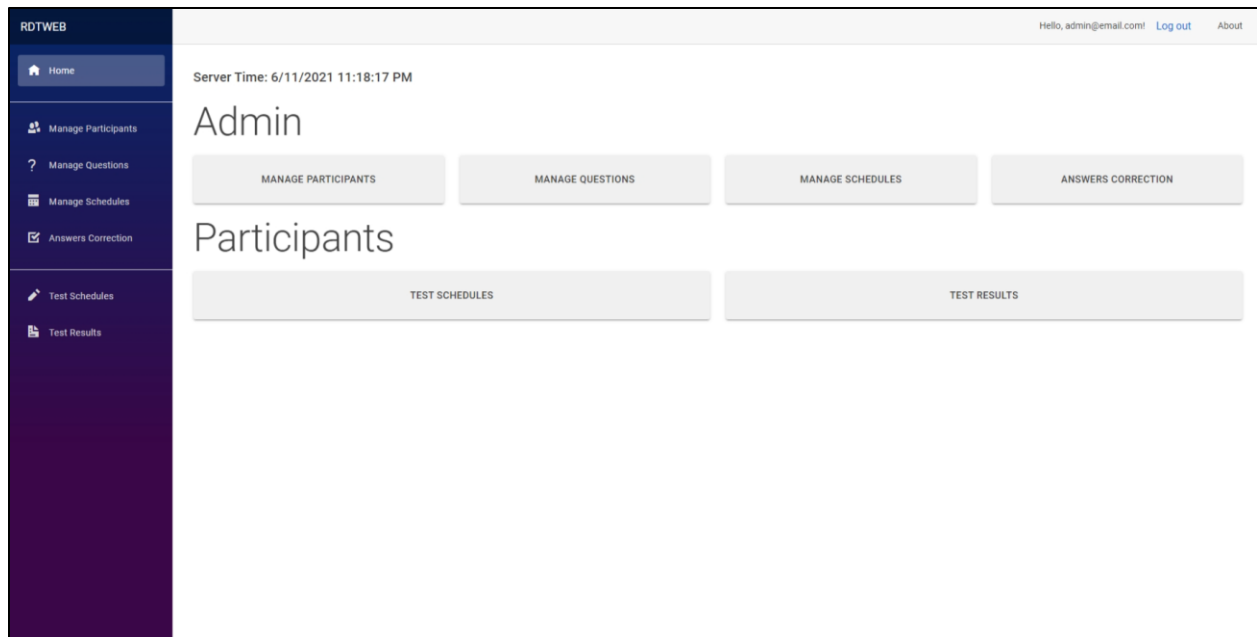


Figure 3. Home (admin).

User can go to navigate to every page from here. This page shows which pages that the user is authorized to visit.

Admin

All pages under “Admin” folder are authorized for users with admin role only.

Manage Participants

The screenshot displays the 'Manage Participants' interface. On the left is a dark sidebar with the 'RDTWEB' logo and navigation links: Home, Manage Participants (active), Manage Questions, Manage Schedules, Answers Correction, Test Schedules, and Test Results. The main content area has a top header with 'Hello, admin@email.com', 'Log out', and 'About'. Below this is a 'Participants' section containing a table with 10 rows. Each row has an ID, an Email address, and a red 'DELETE' button. To the right of the table is a 'Create Participants' form with an 'Email' input field and a 'SUBMIT' button.

ID	Email	Actions
0a9b490f-31ca-4f70-89d0-1dcf950bb05c	joph@email.com	DELETE
4477a405-5bfb-412f-bbe1-7900fa6ee692	clar@email.com	DELETE
5b133abe-0918-40ae-b1c1-ab239132999f	bran@email.com	DELETE
6b167450-82e1-41c1-99c3-4efdz101e95a	onel@email.com	DELETE
839464f3-9cc1-4576-8fd1-3348841872ec	stan@email.com	DELETE
b82bc01c-c3a5-4730-a49a-cfbc00e45bb1	gaby@email.com	DELETE
cba0554c-e08e-4207-9ff6-7b657e424cde	cent@email.com	DELETE
ce4d0032-9356-4193-83cd-068cd5531ffb	cleo@email.com	DELETE

Figure 4. Manage participants.

Admin can view all participants, create, and delete participants in this page.

Manage Questions

Admin can view all questions, view question detail, update and delete questions in this page. Manage Questions are divided into three pages: Index, Create, and View.

Index

RDTWEB		Hello, admin@email.com! Log out About			
Home		Manage Questions			
Manage Participants		Manage Questions			
Manage Questions		Manage Questions			
Manage Schedules		Manage Questions			
Answers Correction		Manage Questions			
Test Schedules		Manage Questions			
Test Results		Manage Questions			

ID	Title	Start	End	Actions
1	Question Set #1	-	-	VIEW DELETE
2	Question Set #2	-	-	VIEW DELETE
3	Question Set #3	-	-	VIEW DELETE
4	Question Set #4	-	-	VIEW DELETE
5	Question Set #5	-	-	VIEW DELETE
6	Question Set #6	6/11/2021 11:02:00 PM	6/12/2021 11:02:00 PM	VIEW DELETE
7	Question Set #7	-	-	VIEW DELETE
8	Question Set #8	-	-	VIEW DELETE
9	Question Set #9	-	-	VIEW DELETE
10	Question Set #10	-	-	VIEW DELETE

Figure 5. Manage questions index.

This page shows all question sets. Question sets can be deleted.

Create

RDTWEB		Hello, admin@email.com! Log out About			
Home		Manage Questions			
Manage Participants		Manage Questions			
Manage Questions		Manage Questions			
Manage Schedules		Manage Questions			
Answers Correction		Manage Questions			
Test Schedules		Manage Questions			
Test Results		Manage Questions			

Question Set Title	Q1
Question Body	Q1
Question Type	Multiple Choice
<input type="checkbox"/> C1	
<input checked="" type="checkbox"/> C2	
<input type="checkbox"/> C3	
<input type="checkbox"/> C4	
REMOVE	
Question Body	Q2
Question Type	Submit File
REMOVE	
+ ADD QUESTION	SAVE BACK

Figure 6. Create question set.

This page shows options for creating question set. Admin can also set question's correct answer. Admin can choose any of the following 5 question types:

- Multiple Choice

User can choose one answer out of four possible answers.

- Boolean (true or false)

User can choose one answer out of two possible answers.

- Multiple Answer List

User can choose one answer out of indefinite possible answers. Admin can indefinite amount of possible answers.

- Essay

User can write multiline string with auto-save feature with delay of 500ms.

- Submit File

User can upload any file.

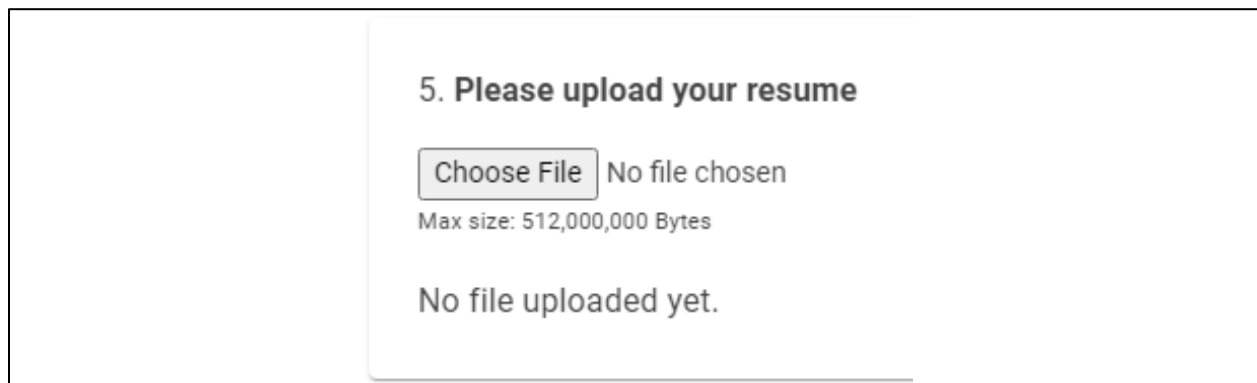


Figure 7. File upload (no file).

5. Please upload your resume

Choose File software compilation by subject.txt
Max size: 512,000,000 Bytes

Name	sffcyrkm.fd1.txt
Size	1,171 Bytes
Last Modified At	6/11/2021 11:50:14 PM

[Download](#)

Figure 8. File upload (uploaded).

View

RDTWEB Hello, admin@email.com! [Log out](#) [About](#)

[Home](#) [Manage Participants](#) [Manage Questions](#) [Manage Schedules](#) [Answers Correction](#) [Test Schedules](#) [Test Results](#)

Question Set #1

1. The correct answer is 1 [Multiple Choice]

☐ One

☒ 1

☐ one

☐ I

[EDIT](#) [DELETE](#)

2. The correct answer is True [Boolean (true or false)]

☒ True ☐ False

[EDIT](#) [DELETE](#)

3. The fifth answer is correct [Multiple Answer List]

☐ Correct

☐ fifth answer

☐ Fifth Answer

☐ wrong

Figure 9. View question set detail.

Admin can view all questions in a question set, update its type, and correct answer, and delete questions.

Manage Schedules

Manage Schedules

ID	Title	Start At	End At	Actions
1	Question Set #1	-	-	EDIT
2	Question Set #2	-	-	EDIT
3	Question Set #3	-	-	EDIT
4	Question Set #4	-	-	EDIT
5	Question Set #5	-	-	EDIT
6	Question Set #6	06/11/2021 11:02 PM	06/12/2021 11:02 PM	SUBMIT CANCEL
7	Question Set #7	-	-	EDIT
8	Question Set #8	-	-	EDIT
9	Question Set #9	-	-	EDIT
10	Question Set #10	-	-	EDIT

Figure 10. Manage schedules.

Admin can edit start and end time for each question sets listed in this page.

Answer Corrections

Answers Correction

Pending Correction Answers

1. Please upload your resume

Name: fxb1aiu.5g5.jpg
Size: 198,904 Bytes
Last Modified At: 6/11/2021 11:35:31 PM
[DOWNLOAD](#)

[MARK ANSWER AS CORRECT](#) [MARK ANSWER AS INCORRECT](#)

2. In 250 words, please tell us about yourself

In honesty and hardwork we strive for excellence

[MARK ANSWER AS CORRECT](#) [MARK ANSWER AS INCORRECT](#)

Corrected Answers

1. Please upload your code

Name: mfc4sser.pnt.js
Size: 1,331 Bytes
Last Modified At: 6/11/2021 11:35:35 PM
[DOWNLOAD](#)

[MARK ANSWER AS CORRECT](#) [MARK ANSWER AS INCORRECT](#)

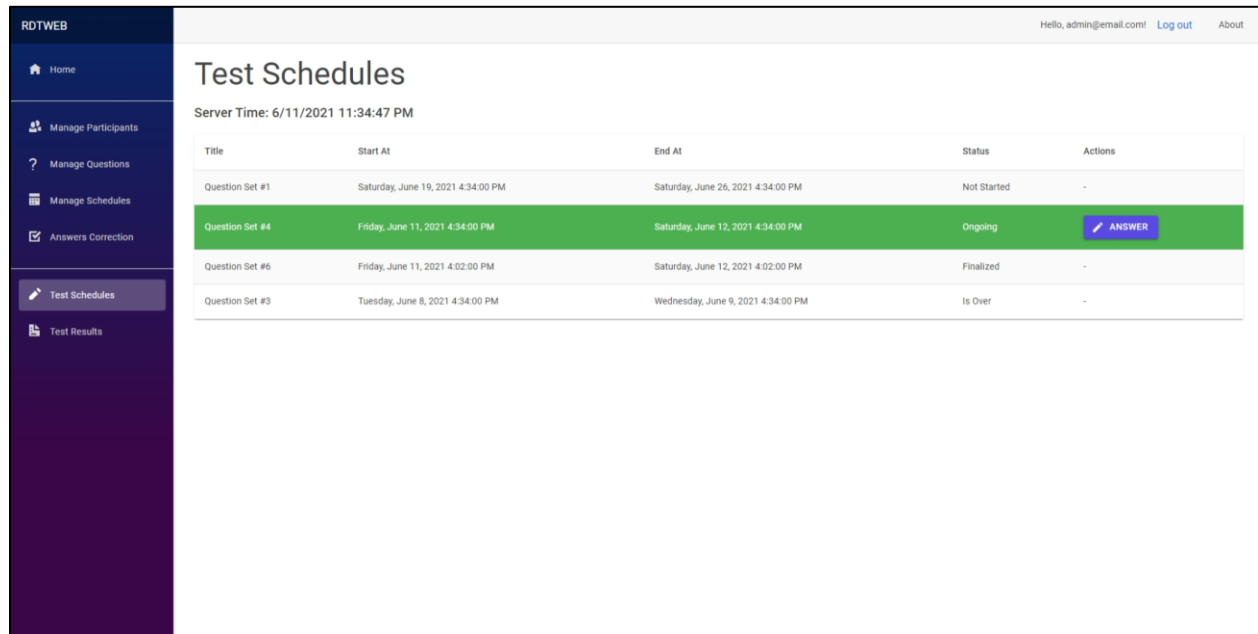
2. Please upload your code

Name: sxxf5e4o.0bs.js
Size: 1,331 Bytes
Last Modified At: 6/11/2021 11:02:30 PM
[DOWNLOAD](#)

Figure 11. Answer corrections.

Admin can mark an answer to be either correct or incorrect. Questions shown here are only of type “Essay” and “Submit File”.

Test Schedules



Title	Start At	End At	Status	Actions
Question Set #1	Saturday, June 19, 2021 4:34:00 PM	Saturday, June 26, 2021 4:34:00 PM	Not Started	-
Question Set #4	Friday, June 11, 2021 4:34:00 PM	Saturday, June 12, 2021 4:34:00 PM	Ongoing	ANSWER
Question Set #6	Friday, June 11, 2021 4:02:00 PM	Saturday, June 12, 2021 4:02:00 PM	Finalized	-
Question Set #3	Tuesday, June 8, 2021 4:34:00 PM	Wednesday, June 9, 2021 4:34:00 PM	Is Over	-

Figure 12. Test schedules.

Users can view all question sets that are either one of “Not Started”, “Ongoing”, “Is Over”, or “Finalized”. Users can attempt to answer an “Ongoing” question set and be taken to “Test Answer” page. Below explains about the test statuses:

- “Finalized” is when the user has finalized their answers in the test.
- “Not started” is when the start date is later than now.
- “Ongoing” is when now is later than start date and now is former than end date.
- “Is Over” is when now is later then end date.

Test Answer

RDTWEB Hello, admin@email.com! [Log out](#) [About](#)

[Home](#)

[Manage Participants](#)

[Manage Questions](#)

[Manage Schedules](#)

[Answers Correction](#)

[Test Schedules](#)

[Test Results](#)

Information

Server Time: 6/11/2021 11:35:51 PM

Title	Question Set #4
Start At	Friday, June 11, 2021 4:34:00 PM
End At	Saturday, June 12, 2021 4:34:00 PM

1. The correct answer is 1

☐ One

☒ 1

☐ one

☐ I

2. The correct answer is True

☒ True ☐ False

3. The fifth answer is correct

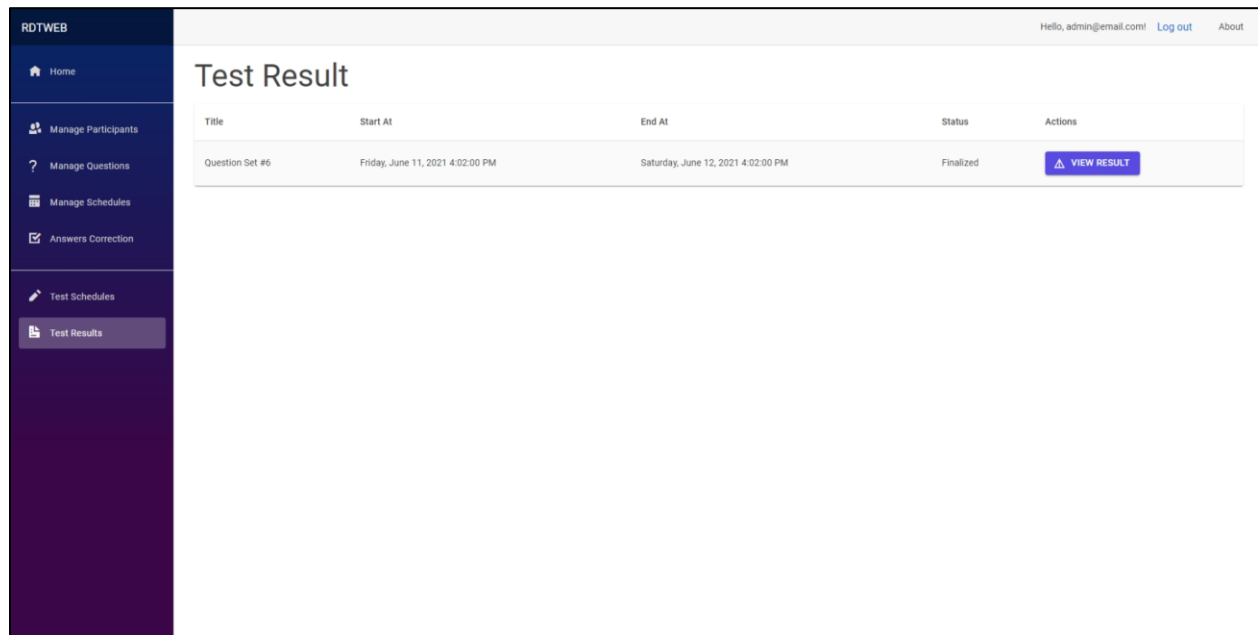
☐ Correct

Figure 13. Test answer.

Users can answer a question set in this page. If a user has finalized their answer for the selected question set, then the user cannot update their answers anymore.

Test Result

Index



The screenshot displays the 'Test Result' index page. On the left is a dark blue sidebar with the 'RDTWEB' logo and a menu including 'Home', 'Manage Participants', 'Manage Questions', 'Manage Schedules', 'Answers Correction', 'Test Schedules', and 'Test Results' (which is highlighted). The top right of the page shows the user 'Hello, admin@email.com!' with links for 'Log out' and 'About'. The main content area is titled 'Test Result' and contains a table with the following data:

Title	Start At	End At	Status	Actions
Question Set #6	Friday, June 11, 2021 4:02:00 PM	Saturday, June 12, 2021 4:02:00 PM	Finalized	VIEW RESULT

Figure 14. Test result index.

User can view question sets that they already answered and finalized. If the user has not finalized their answers, they will not be shown in this page.

View

The screenshot shows the 'Question Set #6' test result view. The interface includes a dark blue sidebar with navigation links: Home, Manage Participants, Manage Questions, Manage Schedules, Answers Correction, Test Schedules, and Test Results (highlighted). The main content area has a header with 'Hello, admin@email.com', 'Log out', and 'About'. Below the header, the title 'Question Set #6' is displayed. A table shows the following statistics:

Correct Answers	3
Pending Corrections	0
Total Questions	6
Final Score	50

Below the table, a green box indicates 'The fifth answer is correct'. It lists six items with checkboxes and their status:

- ☒ Correct
- ☒ fifth answer
- ☒ Fifth Answer
- ☒ wrong
- ☒ wrong answer

At the bottom, a red box prompts the user: 'In 250 words, please tell us about yourself'. Below this prompt is a text input field containing the text 'hehe'.

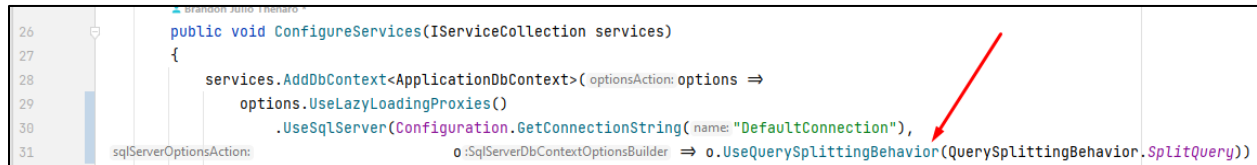
Figure 15. Test result view.

User can view their answers in a question set and view their answer status, that is either correct, incorrect, or pending correction. Grading information is also presented in this page.

Performance Consideration

Split Query

Entity Framework Core's split query¹ feature is used to prevent “cartesian explosion”² problem when querying entities with one-to-many relationship. Hence, the solution is to split the query. The application enables split query globally to fix this problem in “Startup.cs”.

A screenshot of a code editor showing the ConfigureServices method in Startup.cs. The code is as follows:

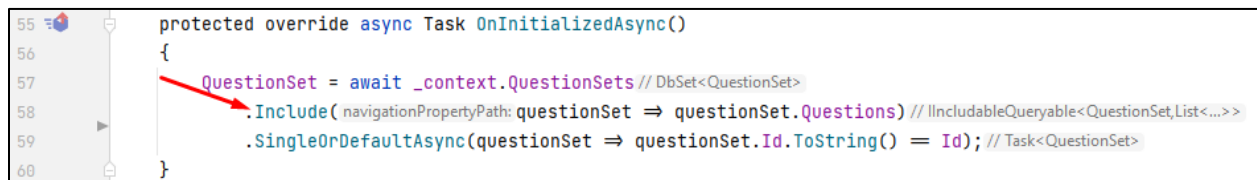
```
26 public void ConfigureServices(IServiceCollection services)
27 {
28     services.AddDbContext<ApplicationDbContext>(optionsAction: options =>
29         options.UseLazyLoadingProxies()
30         .UseSqlServer(Configuration.GetConnectionString("DefaultConnection"),
31             sqlServerOptionsAction: o => o.UseQuerySplittingBehavior(QuerySplittingBehavior.SplitQuery));
```

A red arrow points from the text "split query" in the paragraph above to the `SplitQuery` parameter in the `UseQuerySplittingBehavior` call.

Figure 16. Split query configuration in "Startup.cs".

Eager Loading

Relationships are eagerly loaded and explicitly specified to only load data which are required. Lazy loading is also used but sparingly to prevent N+1 problem, e.g., while looping³.

A screenshot of a code editor showing the OnInitializedAsync method. The code is as follows:

```
55 protected override async Task OnInitializedAsync()
56 {
57     QuestionSet = await _context.QuestionSets // DbSet<QuestionSet>
58     .Include(navigationPropertyPath: questionSet => questionSet.Questions) // IQueryable<QuestionSet, List<...>>
59     .SingleOrDefaultAsync(questionSet => questionSet.Id.ToString() == Id); // Task<QuestionSet>
60 }
```

A red arrow points from the text "explicit eager loading" in the caption below to the `.Include` call on line 58.

Figure 17. An example of explicit eager loading in "ManageQuestions/Index.razor".

Virtualize Component

Virtualize component is built-in Blazor component to improve UI performance while rendering list by limiting UI rendering only to parts that are currently visible⁴.

¹ [Single vs. Split Queries - EF Core | Microsoft Docs](#)

² [The performance issue "Cartesian Explosion" made its comeback in Entity Framework Core 3 - Thinkecture](#)

³ [Efficient Querying - EF Core | Microsoft Docs](#)

⁴ [ASP.NET Core Blazor component virtualization | Microsoft Docs](#)



```

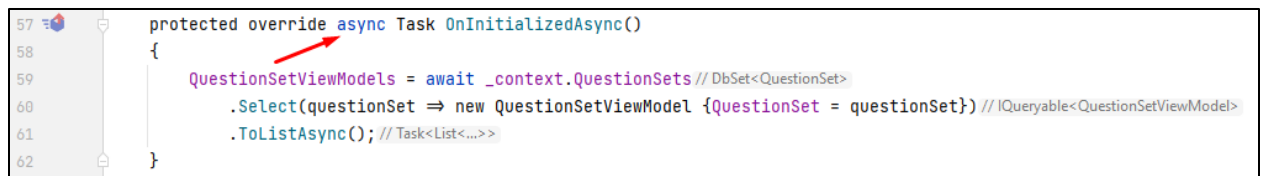
25 <Virtualize Items="Users">
26 <tr>
27 <td class="align-middle">@context.Id</td>
28 <td class="align-middle">@context.Email</td>
29 <td class="align-middle">
30 <MudButton OnClick="() => DeleteUser(context)" Variant="Variant.Filled" Color="Color.Error" StartIcon="@Icons.Material.Filled.Delete">
31 Delete
32 </MudButton>
33 </td>
34 </tr>
35 </Virtualize>

```

Figure 18. `<Virtualize />` example in "ManageParticipants.razor".

Asynchronous Programming

Asynchronous programming is applied while coding this application to prevent blocking the UI thread⁵ so that the UI is responsive to user's input and feels snappy.



```

57 protected override async Task OnInitializedAsync()
58 {
59     QuestionSetViewModels = await _context.QuestionSets // DbSet<QuestionSet>
60     .Select(questionSet => new QuestionSetViewModel {QuestionSet = questionSet}) // IQueryable<QuestionSetViewModel>
61     .ToListAsync(); // Task<List<...>>
62 }

```

Figure 19. Asynchronous programming example in "ManageQuestions/Index.razor".

⁵ [Asynchronous Programming - EF Core | Microsoft Docs](#)

Notes

- The .NET version used while developing the app is .NET 5.0.301.
- Admin account is admin@email.com with password “d3F@uLtP@55w0Rd”.
- Please relog after seeding to avoid errors while answering questions.
- Every created user has default password of “d3F@uLtP@55w0Rd”.
- Default password can be changed in “appsettings.json”.
- Run seeding on every launch can be configured in “appsetting.json”.
- Used DBMS is SQL Server with database named “rdt” and uses Windows Authentication.

ERD

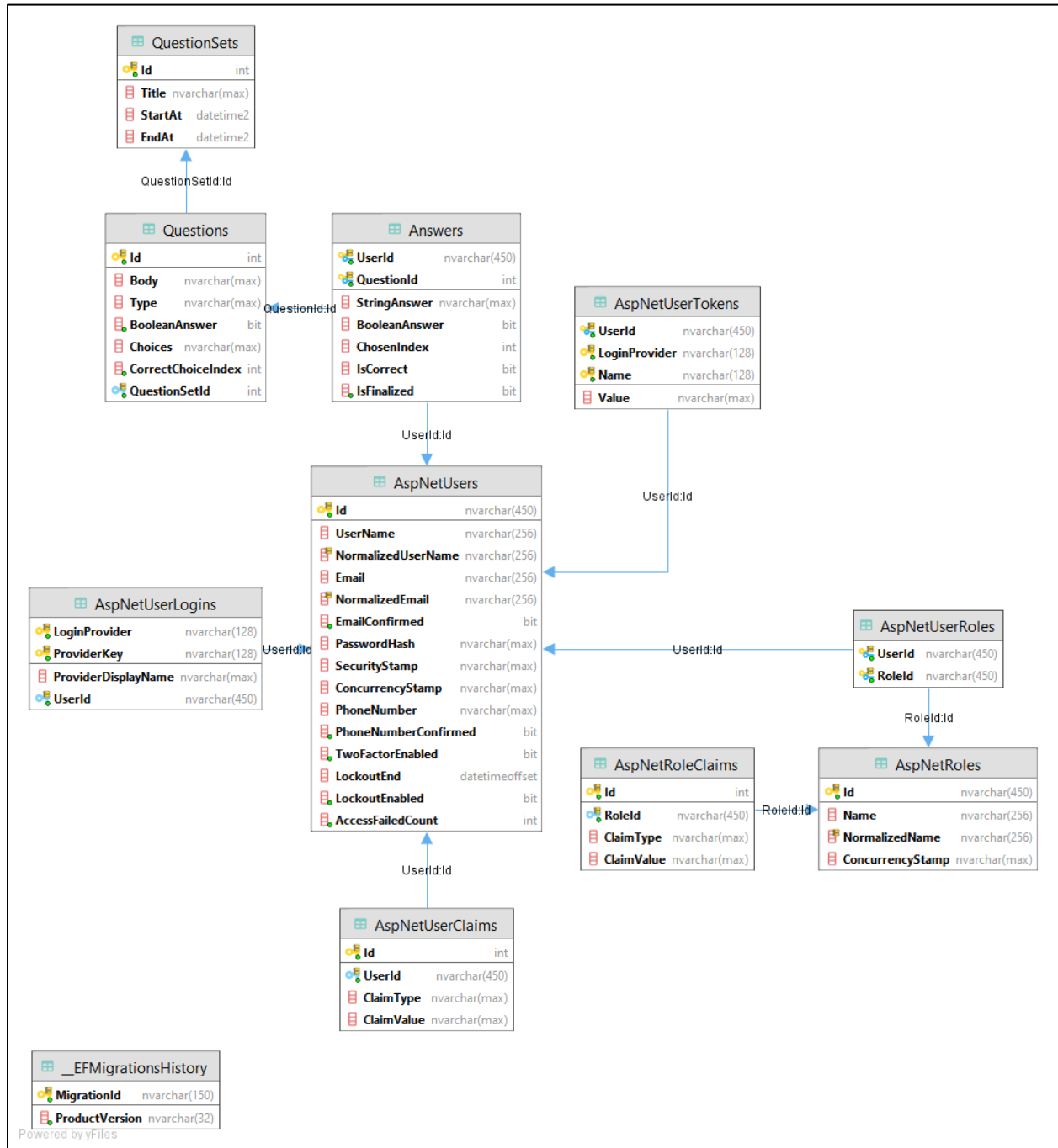


Figure 20. ERD

References

[ASP.NET Web Apps | HTML, CSS, JavaScript, and C# \(microsoft.com\)](#)

[Choose between ASP.NET 4.x and ASP.NET Core | Microsoft Docs](#)

[Introduction to Identity on ASP.NET Core | Microsoft Docs](#)

[Blazor | Build client web apps with C# | .NET \(microsoft.com\)](#)

[ASP.NET Core Blazor hosting models | Microsoft Docs](#)

[Overview of Entity Framework Core - EF Core | Microsoft Docs](#)

[Introduction · Bootstrap \(getbootstrap.com\)](#)

Table of Contents

Foreword	2
Technologies	3
ASP.NET Core 5.0	3
ASP.NET Core Identity	3
Blazor (Server)	3
Entity Framework Core + SQL Server	3
Bootstrap 4.3.1	3
Pages	4
Admin	4
Manage Participants	6
Manage Questions	6
Manage Schedules	10
Answer Corrections	10
Test Schedules	11
Test Answer	12
Test Result	13
Index	13
View	14
Performance Consideration	15

Split Query	15
Eager Loading	15
Virtualize Component	15
Asynchronous Programming	16
Notes	17
ERD.....	18
References	19
Table of Contents	20