

# Ashak Zahin Hasan

Address: East Shewrapara, Mirpur, Dhaka – 1216

Phone: +8801515667288

Website: <https://aboutme-b0750.web.app/>

Email: [azahinhasan@gmail.com](mailto:azahinhasan@gmail.com)

GitHub: <https://github.com/azahinhasan>

---

## EDUCATION

- **American International University-Bangladesh (AIUB),**  
Dhaka— Bachelor of Science Degree in Computer Science and Engineering  
Jan 2018 - Present (Current 11<sup>th</sup> Semester)
  - **Cambrian School & College,**  
Dhaka— Higher Secondary Certificate  
Feb 2015- April 2017
  - **Mirpur Bangla School,**  
Dhaka—Secondary School Certificate  
Jan 2013- Jan 2015
- 

## AWARDS

- Academic Dean's Awards, Spring 2018-19
- 

## SKILLS

**Programing Language:** JavaScript, PHP, C#, HTML, CSS, Python, Java, C/C++

**Framework:** React.JS, Redux, ASP.Net, jQuery, Laravel, React Native, Bootstrap

**Other:** Algorithm/Data-structure, Usages of SqlDB, Restful API, Google Firebase

# PROJECTS

## **-Laptop Shop WebApp, React.js, ASP.Net and MsSql**

### Description:

This one is the largest project so far, I have done myself. It's a full-stack Webapp with Restful API. There are 2 main types of users. Which are Customer and Employee.

Customer: They can (also employees) search for any product and add them to the cart with quantity. Then they can Order also they can use Promo codes it will give them some offers if it's a valid code. They can confirm the order with or without Login. And print the receipt. If the customer is registered and done this order by login then order information will store in the Database and he can check all orders and order's current progress status and comment from employees if there is any in their Orders Page. From this page, he/she can also cancel the Order if the status is 'none (no action take yet)'.

Employee: Three types in Employee: Admin, Moderator, Delivery Manager. Moderator and, Delivery Manager has some restriction. By default, the Delivery Manager can see all orders and update the progress status of orders (if the order is not canceled) and put a comment. Moderators can add, edit Product, and Promo code information. Admin can add new employees, modify access of any employee and modify salary info including the above options. And more.

Every user can be stay login the same time in one browser. If he login from another browser the previous one will log out auto. And there are more features.

### GitHub:

<https://github.com/azahinhasan/PcHardwareShop>

## **- Wi-Fi Info QR-code Generator, React.js and CSS**

### Description:

Users can generate QR-Code with Wi-Fi information. By scanning the QR anyone can connect with that Wi-Fi network. The QR code is downloadable as PNG or PDF. There is a Dark and Light mood for this web app.

### GitHub:

[https://github.com/azahinhasan/WiFi\\_Info\\_QRcode\\_Generator](https://github.com/azahinhasan/WiFi_Info_QRcode_Generator)

### Live View:

[https://azahinhasan.github.io/WiFi\\_Info\\_QRcode\\_Generator/](https://azahinhasan.github.io/WiFi_Info_QRcode_Generator/)

## **-Blood Donation WebApp, jQuery, ASP.Net and MsSql**

### Description:

This is a group project. There are two projects on the topic.

1. In backend we used Code Fast method (with auto migration). In frontend we are accessing data by API HTTP request and used jQuery. There are three users Admin, Donner, and User (who are seeking for donner). I did the Admin and Moderator part.

In this part admin can add a new employee and print the receipt, edit salary info, Promotion/Demotion Employee, disable the account, check all Reports that are made by Users and reviews them if use got too many reports, then Admin/Moderator can Ban/Unban that user, reviews the message of all user which are done from Contact us page and more.

User (donner seeker) and Donner can do some more things. Users can search for donner and request them for detention. The notification will go to Donner if he accepts it User will get it to know from Notification.

Both can edit their profile info. Anyone can make some donations for our site and more.

2. This one has almost the same feature as the Above one. But this one is made without Restful API and Frontend Framework. Here, we made it by following MVC architecture and DB Fast Method by using only the Dot Net framework.

### 1.GitHub:

[https://github.com/azahinhasan/BloodDonation\\_AspDotNetjQuery](https://github.com/azahinhasan/BloodDonation_AspDotNetjQuery)

### 2.GitHub:

<https://github.com/azahinhasan/BloodDonationProject>

## **- Weather WebApp, React.js and CSS**

### Description:

Users can find out the current Weather info from this WebApp. I take help from two open-source API. Which Provide me the Weather info. User Can search the location or there is a find out location option which will detect the current location automatically.

### GitHub:

[https://github.com/azahinhasan/weatherApp\\_React](https://github.com/azahinhasan/weatherApp_React)

### Live View:

[https://azahinhasan.github.io/weatherApp\\_React/](https://azahinhasan.github.io/weatherApp_React/)

**- Anime Info Mobile App, React Native**

Description:

This app will provide the user information about Anime (Japanese Cartoon) and Manga (Japanese Comics). I used API for information. There are two main Scene Option one is for Anime other one is for Manga. On the Anime screen, there is a Search option Where the user can search any Anime and it will provide the result according to the search. There is the Top Anime part where at default there will be a top 50 results but users can see more by tapping on Arrow key. Every time tapping on the arrow key will add more 50 results with previous results. And their Upcoming part. There will be a list of Summer, Winter, and Fall upcoming anime. By tapping on any specific anime name or picture it will take to the Details screen. There will be all info about that anime and Genres part. By tapping on any genres name it will take to the screen where it will show the list related to that genre. The manga part is the same as the above Anime part.

GitHub:

[https://github.com/azahinhasan/AnimeInfoApp\\_React\\_Native](https://github.com/azahinhasan/AnimeInfoApp_React_Native)

Download Link (APK only):

[https://github.com/azahinhasan/AnimeInfoApp\\_React\\_Native/releases/tag/Beta](https://github.com/azahinhasan/AnimeInfoApp_React_Native/releases/tag/Beta)

**- Covid19 Tracker WebApp, React.js and CSS**

Description:

Users can find out the Covid-19 statistics. There is a World Statistics part and Country part when the user can search the country. There is a List page where users can find the list of 220+ country's data. Also, they can sort the data by Population, Cases, Recovered, etc. By clicking on the specific Table Header or choosing from the Sort by drop-down option.

GitHub:

[https://github.com/azahinhasan/covid19\\_tracker\\_React.js](https://github.com/azahinhasan/covid19_tracker_React.js)

Live View:

[https://azahinhasan.github.io/covid19\\_tracker\\_React.js/#/home](https://azahinhasan.github.io/covid19_tracker_React.js/#/home)

**- To-do List with Auth, React.js, Redux, Firebase and CSS**

Description:

It's a To-Do list web app. Where users can delete, add or mark as done their to-dos.

But the Interesting part is users have to Sign-In/Sign-Up fast. There are 3 options for these two actions. Which are the Normal Manual user Login option where have to Sign-Up for this one, other two are Sign-In with Google or Facebook [user can connect their Google or Facebook account directly]. Using Redux for accessing the Middleware to do the user authentication in there. And storing the all-user dates in Google Firebase.

GitHub:

[https://github.com/azahinhasan/todoListV2\\_React\\_withAuth](https://github.com/azahinhasan/todoListV2_React_withAuth)

Live View:

[https://azahinhasan.github.io/todoListV2\\_React\\_withAuth/#/SignIn](https://azahinhasan.github.io/todoListV2_React_withAuth/#/SignIn)

**- GuessTheNumber\_V2, Vanilla JavaScript, HTML and CSS**

Description:

It is a small game where the player has to identify the number from the option. Easy to win if anyone catches a logic. There is a Hints option where I will give a random range of numbers. But using Hints will cost -3 from the total points. For right guess, the player will get +5. For the wrong guess, the player will lose 1 life. There will be a total of 5 Life. To win players have to manage total points of more than 29. If the point is less than -10 or life is less than 0 game over.

GitHub V2:

[https://github.com/azahinhasan/GuessTheNumber\\_V2](https://github.com/azahinhasan/GuessTheNumber_V2)

Live View V2:

[https://azahinhasan.github.io/GuessTheNumber\\_V2/](https://azahinhasan.github.io/GuessTheNumber_V2/)

Live View V1:

<https://azahinhasan.github.io/GuessTheNumber/>

<p><b>- Download Time Calculator</b>, <u>Vanilla JS, HTML and CSS</u></p> <p><u>Description:</u> User have to provide Size of the file (MB/GB) and Net Speed (MBps/Mbps) then user will get the total time that it will take to complete the download.</p> <p><u>GitHub:</u> <a href="https://github.com/azahinhasan/downloadTimeCalculator">https://github.com/azahinhasan/downloadTimeCalculator</a></p> <p><u>Live View:</u> <a href="https://azahinhasan.github.io/downloadTimeCalculator/">https://azahinhasan.github.io/downloadTimeCalculator/</a></p>	<p><b>- Word Counter</b>, <u>Vanilla JS, HTML and CSS</u></p> <p><u>Description:</u> This site will show the number of Total words, Characters and Sentences of a paragraph or sentence or word.</p> <p><u>GitHub:</u> <a href="https://github.com/azahinhasan/wordCounter">https://github.com/azahinhasan/wordCounter</a></p> <p><u>Live View:</u> <a href="https://azahinhasan.github.io/wordCounter/wordCounter">https://azahinhasan.github.io/wordCounter/wordCounter</a></p>
<p><b>- Coffee Shop Management Windows App</b>, <u>C# (with Windows Form) and Microsoft SQL Server</u></p> <p><u>Description:</u> It is a group project. Make by following MVC architecture. There are 4 types of main users. And There are the Admin, Customer, Cook, and Receptionist. Admin can add/delete/edit Coffee, modify Vat, add new Employee, modify salary, and more. Customers can order and see the specific item and total cost. The Receptionist can take the order and send the data to Cook. Cook will check it and prepare the order. After done he can mark the order as done. And there is more feature.</p> <p><u>GitHub:</u> <a href="https://github.com/azahinhasan/BloodDonationProject">https://github.com/azahinhasan/BloodDonationProject</a></p>	<p><b>- Tutor Finding Website</b>, <u>PHP, HTML, CSS, Ajax, JS and XAMPP SQL Server</u></p> <p><u>Description:</u> There are 3 types for users: Admin, Tutor, and Tutor seeker. Admin modifies information regarding Employees. Such as add/edit/remove/see the list/change rank of employees, verify Tutor, review contact us texts, etc. The tutor can post a blog, edit Info, accept/reject Tutor seeker requests etc. Tutor seekers can search tutors, send them requests for teaching, make payments etc.</p> <p><u>GitHub:</u> <a href="https://github.com/azahinhasan/tutorFinding_Final">https://github.com/azahinhasan/tutorFinding_Final</a></p>
<p><b>- Set turnoff/restart/lock/sleep timer for Pc</b>, <u>Python</u></p> <p><u>Description:</u> Timer for turn-off/restart/lock/sleep user PC. It's just a console program. Users have to choose the action fast then set a time to execute the action.</p> <p><u>GitHub:</u> <a href="https://github.com/azahinhasan/turnOFF_restart_lock_sleep_pc">https://github.com/azahinhasan/turnOFF_restart_lock_sleep_pc</a></p>	<p><b>- Prison Management System</b>, <u>Java and XAMPP SQL server</u></p> <p><u>GitHub:</u> <a href="https://github.com/azahinhasan/Prison_Management_System">https://github.com/azahinhasan/Prison_Management_System</a></p>