

Rancang Bangun *Platform Competitive Programming* Sebagai Media Belajar Struktur Data dan Algoritma Berbasis Web

Muhammad Al Farizzi*, Dwi Amalia Purnamasari*

* Informatics Engineering, Batam State Polytechnic

alfarizzi2711@gmail.com¹, dwiamalia@polibatam.ac.id²

Article Info

Article history:

Received Jun 12th, 201x

Revised Aug 20th, 201x

Accepted Aug 26th, 201x

Keyword:

Competitive Programming
Data Structure & Algorithm
PHP
Laravel

ABSTRACT

The design of a competitive programming platform as a medium for learning web-based data structures and algorithms is research carried out to build a platform that can be used to work on data structure and algorithm problems. This research was carried out using the waterfall method, the final result of this research is a web that can be used to work on questions which are expected to help improve fundamental understanding of data structures and algorithms which are the main foundation that is very important in making software. The conclusion of this research is the importance of understanding the fundamentals of data structures and algorithms in order to be able to create good software. This understanding can be improved by practicing a lot of questions so that we can use the right data structure based on the case at hand.

Copyright © 201x Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

1. PENDAHULUAN

Digitalisasi adalah bentuk proses perubahan dari penggunaan analog ke teknologi digital, dengan melakukan digitalisasi data dan informasi, maka data serta informasi tersebut tidak hanya terselamatkan, namun juga dapat disebar kemudian diakses oleh siapa saja, dimana saja, dan kapan saja. Digitalisasi dapat berupa sebuah sistem informasi berbasis web maupun aplikasi, dimana keduanya dapat dijalankan diberbagai media. Digitalisasi ini umumnya dilakukan oleh orang yang menguasai teknologi informasi atau lebih dikenal dengan teknik informatika. Dalam era digitalisasi, mahasiswa teknik informatika harus dipersiapkan dengan pemahaman yang mendalam terhadap konsep-konsep fundamental algoritma dan struktur data. Kedua konsep ini merupakan pondasi utama dalam pengembangan perangkat lunak dan pemecahan masalah di dunia nyata. Namun, pengajaran yang saat ini marak dilakukan hanya berfokus pada teori yang seringkali tidak mencukupi untuk mengasah kemampuan praktis mahasiswa.

Pentingnya keahlian dalam algoritma dan struktur data semakin terasa terutama karena industri teknologi informasi semakin menuntut kecepatan dan efisiensi dalam pengembangan perangkat lunak. Mahasiswa tidak hanya perlu memahami konsep-konsep tersebut secara teoritis, tetapi juga mampu mengimplementasikan dalam situasi dunia nyata. Salah satu pendekatan yang bisa dilakukan untuk meningkatkan pemahaman praktis adalah melalui *competitive programming*. *Competitive Programming* adalah praktik *coding* untuk merancang solusi dari sebuah masalah secara *competitive*. *Code* yang ditulis dinilai berdasarkan berbagai faktor yang mencakup efisiensi dan keakuratan *code* tersebut [1]. *Competitive Programming* menawarkan konteks yang menantang dan kompetitif memaksa mahasiswa untuk menerapkan struktur data dan algoritma.

Dapat disimpulkan dengan permintaan industri teknologi informasi terhadap kecepatan dan efisiensi dalam pengembangan perangkat lunak, menjadi jelas bahwa mahasiswa teknik informatika harus memiliki pemahaman praktis yang kuat tentang konsep-konsep fundamental algoritma dan struktur data. Keduanya merupakan pondasi utama dalam proses pengembangan perangkat yang efektif daana efisien. Algoritma dan struktur data tidak hanya penting untuk memahami bagaimana suatu program bekerja, tetapi juga untuk mengoptimalkan kinerja program dan memecahkan masalah dengan cara yang efisien. Salah satu solusi yang dapat diterapkan adalah dengan membangun *platform* yang dapat digunakan sebagai sarana belajar struktur data dan algoritma dengan mengerjakan soal *competitive programming*. Diharapkan *platform* ini dapat

menjadi wadah bagi mahasiswa untuk menjalani pengalaman belajar melalui *competitive programming, platform* yang dikembangkan bertujuan tidak hanya sebagai alat evaluasi, tetapi juga sebagai sarana pembelajaran interaktif yang mendalam, *platform* ini juga diharapkan dapat menjadi laboratorium *virtual* bagi mahasiswa untuk menguji dan meningkatkan pemahaman mereka tentang algoritma dan struktur data.

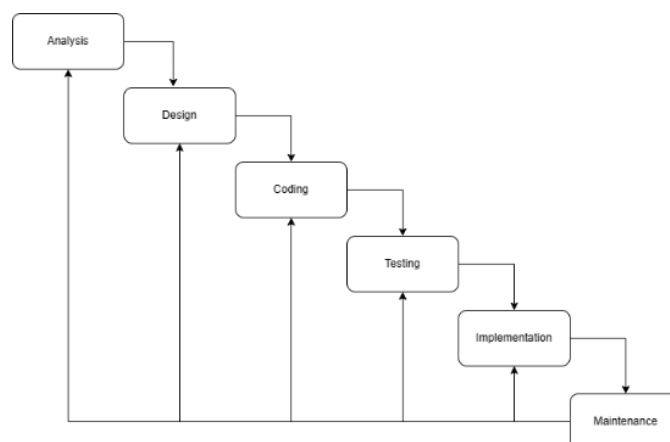
2. TINJAUAN PUSTAKA

Berdasarkan penelitian yang berjudul “Pengembangan Game Edukasi Dengan Metode GDLC: Studi Kasus Mata Kuliah Algoritma dan Struktur Data”. Dengan adanya media belajar algoritma dan struktur data berupa game dapat memberikan pemahaman yang lebih mudah kepada pengguna [2]. Berdasarkan penelitian yang berjudul “Implementasi Augmented Reality (AR) Sebagai Sarana Media Pembelajaran Interaktif Algoritma Pemrograman”. Penggunaan teknologi AR sangat penting sekali sebagai sarana pembelajaran secara interaktif dan inovatif karena peserta didik dapat melihat secara langsung di dunia nyata dan dapat berimajinasi sehingga mampu meningkatkan minat untuk belajar bagi peserta mencapai 80% [3]. Berdasarkan penelitian yang berjudul “Perancangan Media Pembelajaran Berbasis Audio Visual Pada Materi Looping Mata Kuliah Algoritma dan Struktur Data Universitas Gorontalo”. Dengan adanya media belajar berbasis audio visual dapat menunjang keberhasilan mahasiswa dalam mencapai kompetensi yang diajarkan sehingga dapat menumbuhkan antusias dan motivasi belajar bagi mahasiswa pada mata kuliah algoritma dan struktur data [4]. Berdasarkan penelitian yang berjudul “Pembangunan Aplikasi *Reality* Terimbuh(VR) Untuk Pengenalan Struktur Data”. Dengan adanya aplikasi *reality* terimbuh(VR) dapat mempermudah pelajar dalam memahami konsep *linked list & pointer* [5]. Berdasarkan penelitian yang berjudul “Perancangan *Intelligent Tutoring System* Sebagai Upaya Inovatif Pada Pembelajaran Algoritma dan Struktur Data”. Dengan adanya sistem ini dapat membantu mahasiswa untuk belajar dengan lebih efektif dan efisien [6].

Penelitian – penelitian sebelumnya telah menghasilkan berbagai produk seperti *Augmented Reality (AR)*, *Game*, dan *Intelligent Tutoring System* berbasis *website*. Penelitian – penelitian ini memiliki tujuan serupa, yaitu membantu proses pembelajaran struktur data dan algoritma. Meskipun penelitian – penelitian sebelumnya cenderung menyajikan penjelasan dan konten yang mudah dimengerti oleh mahasiswa, namun kurang memberikan pengalaman praktik langsung dalam penggunaan struktur data untuk menyelesaikan masalah. Penelitian ini diharapkan dapat mengisi kesenjangan tersebut dengan memberikan pengalaman praktik kepada mahasiswa. Fokus utama adalah memberikan kesempatan kepada mahasiswa untuk secara aktif menggunakan struktur data yang telah dipelajari dari mata kuliah struktur data dalam menyelesaikan soal algoritmik. Dengan demikian, penelitian ini bertujuan untuk melengkapi pemahaman teoritis dengan pemahaman praktis yang diharapkan dapat meningkatkan kemampuan mahasiswa dalam menerapkan konsep-konsep struktur data dalam situasi dunia nyata. Diharapkan bahwa pendekatan ini tidak hanya memberikan pemahaman konsep yang kuat, tetapi juga memperkaya pembelajaran mahasiswa melalui pemahaman langsung. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi positif terhadap pembelajaran mahasiswa dalam konteks struktur data dan algoritma.

3. METODOLOGI REKAYASA

Metode yang digunakan dalam perancangan dan pembangunan *platform* ini adalah *waterfall*, *waterfall* adalah model pengembangan perangkat lunak yang sistematis dan berurutan [7].

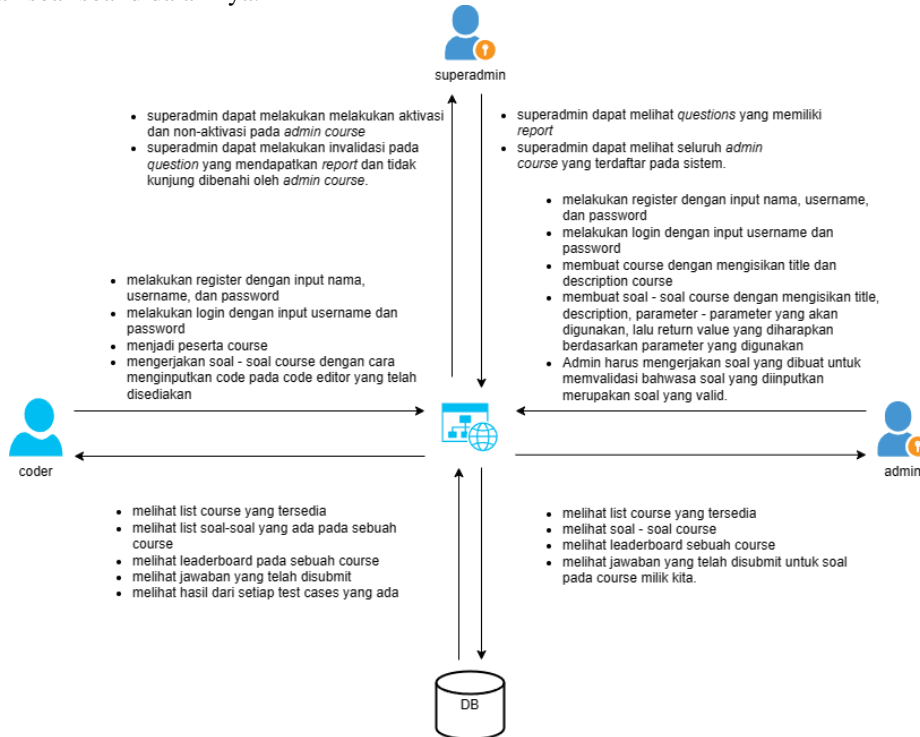


Gambar 1: Metode Waterfall [8]

Metode ini terdiri dari 6 tahap, yaitu:

3.1. Analysis

Tahapan ini dilakukan dengan pengumpulan data dengan cara melakukan riset pada 5 jurnal yang berkaitan dengan pengembangan sistem untuk pembelajaran struktur data. Berdasarkan hasil riset yang telah dilakukan dihasilkan gambaran umum dari *platform* yang akan dibuat sesuai dengan Gambar 2, dimana terdapat 3 aktor yaitu *superadmin*, *admin*, dan *coder*. *Superadmin* adalah aktor yang bertugas untuk mengawasi *platform* dimana *superadmin* dapat melakukan aktivasi / non-aktivasi akun dari *admin* juga, dan juga dapat menetapkan soal menjadi tidak valid jika sudah banyak laporan yang masuk dari para *coder*. *Admin* adalah aktor yang bisa membuat *course* yang bisa diikuti oleh para *coder*, dimana pada *course* ini akan terdapat soal-soal yang dapat dikerjakan. *Coder* adalah aktor yang bisa bergabung pada *course* dan mengerjakan soal-soal didalamnya.



Gambar 2: Gambaran Umum Sistem

Berdasarkan gambaran umum yang telah dihasilkan maka didapati kebutuhan fungsional dan non-fungsional dari *platform* yang akan dibangun sesuai Tabel 1 untuk kebutuhan fungsional dan Tabel 2 untuk kebutuhan non-fungsional.

Tabel 1: Kebutuhan Fungsional

No	Kebutuhan Fungsional
F01	Admin dapat melakukan registrasi
F02	Admin dapat melakukan login
F03	Admin dapat membuat sebuah course
F04	Admin dapat menambahkan soal-soal kedalam course
F05	Admin dapat menghapus soal dari course
F06	Admin dapat melihat leaderboard sebuah course
F07	Admin dapat melihat peserta course
F08	Admin dapat melihat jawaban yang telah di-inputkan oleh coder
F09	Admin dapat menambahkan parameter yang akan digunakan untuk testing dan hasil yang diharapkan untuk soal yang dibuat
F10	Admin dapat mengerjakan soal yang dibuat untuk membuktikan bahwa soal tersebut valid dan dapat dikerjakan
F11	Coder dapat melakukan registrasi
F12	Coder dapat melakukan login
F13	Coder dapat berpartisipasi dalam sebuah course

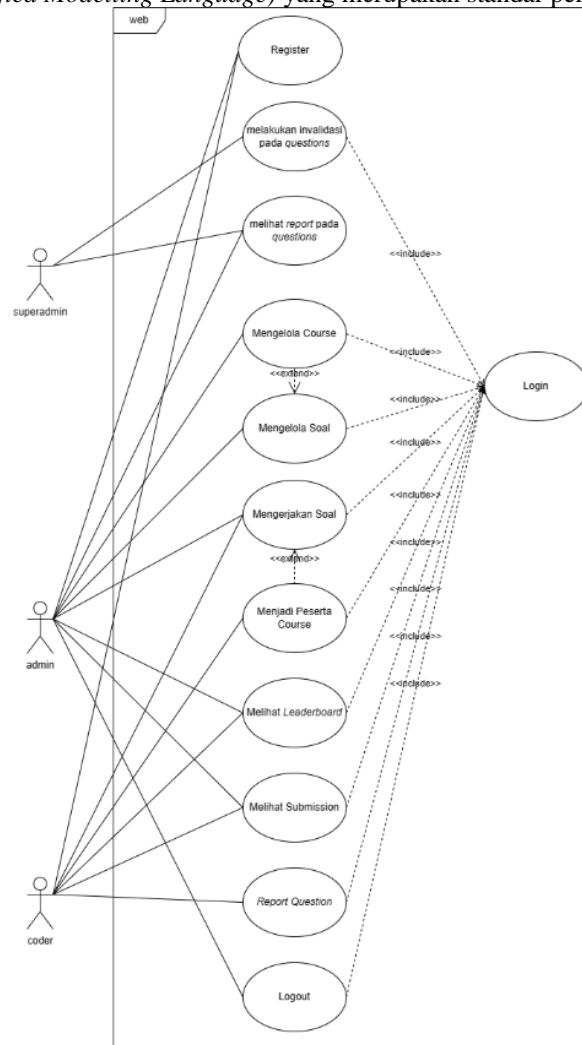
No	Kebutuhan Fungsional
F14	Coder dapat mengerjakan soal dengan <i>code editor</i> yang ada pada
F15	Coder dapat melihat <i>testcases</i> yang berhasil dan gagal
F16	Coder dapat melihat <i>leaderboard</i> pada <i>course</i>
F17	Coder dapat melihat jumlah soal yang telah berhasil dikerjakan pada sebuah <i>course</i>
F18	<i>Superadmin</i> dapat melakukan login
F19	<i>Superadmin</i> dapat melakukan validasi pada profil <i>admin</i> <i>course</i>

Tabel 2: Kebutuhan Non-Fungsional

No	Kebutuhan Non Fungsional
NF01	Bahasa pemrograman yang digunakan adalah <i>PHP</i> dengan <i>framework Laravel</i>
NF02	<i>Database</i> yang digunakan adalah <i>PostgreSQL</i>
NF03	<i>Web</i> bisa di- <i>deploy</i> dengan <i>container deployment</i>

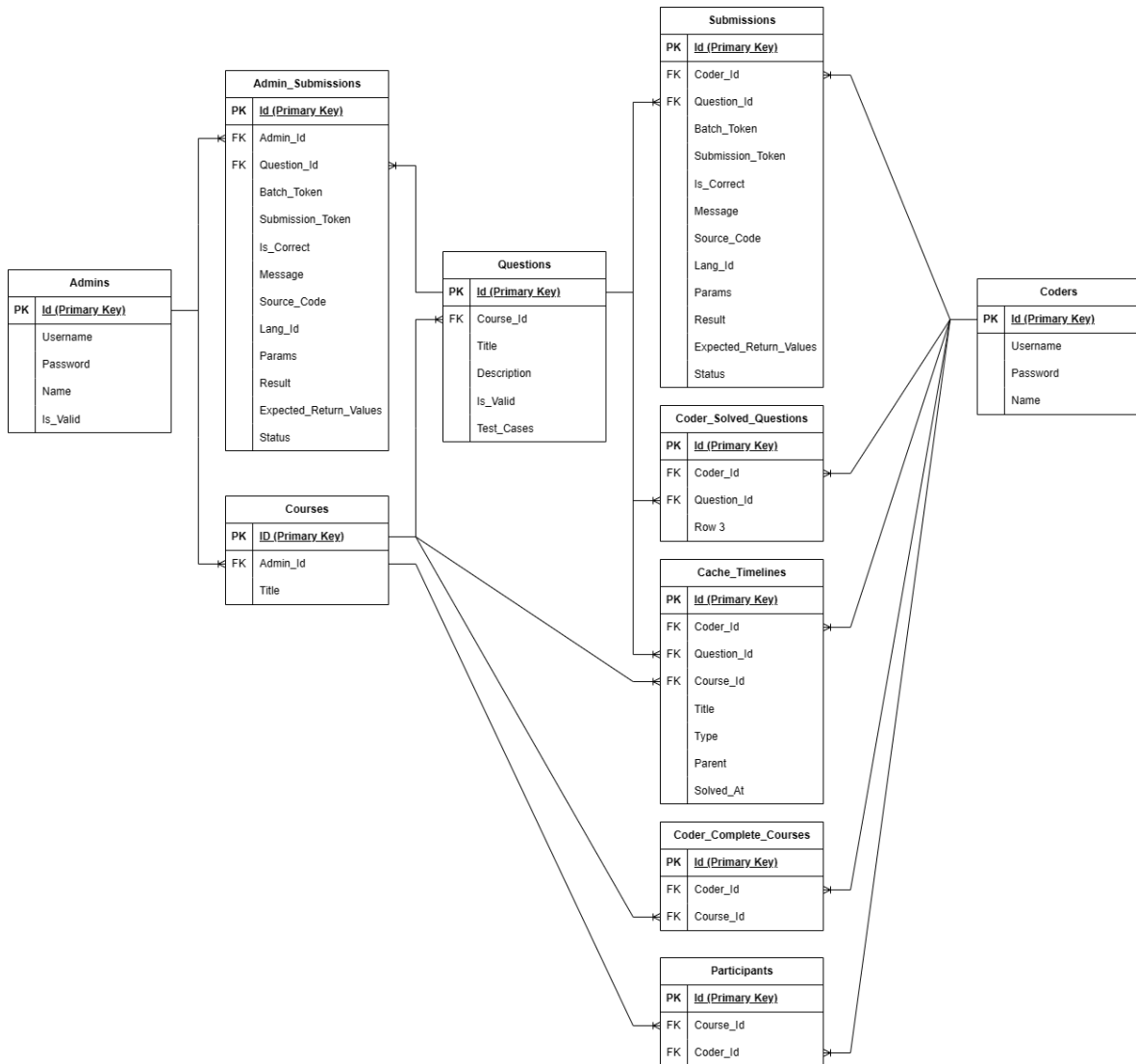
3.2. Design

Pada tahap ini peneliti melakukan rancangan pada skema relasi tabel yang berguna sebagai panduan untuk membuat tabel untuk menyimpan data, dan *usecase diagram* yang berguna untuk menggambarkan aksi yang dapat dilakukan masing-masing actor. *Usecase diagram* yang digunakan untuk menggambarkan aksi yang dapat dilakukan masing-masing aktor sesuai dengan Gambar 3. etode yang digunakan untuk pemodelan *usecase* adalah *UML (Unified Modelling Language)* yang merupakan standar pemodelan secara visual [9].



Gambar 3: Usecase Diagram Platform Competitive Programming

Pada *platform* ini terdapat 3 aktor sesuai dengan Gambar 3 yaitu *superadmin*, *coder*, dan *admin*. *Superadmin* adalah aktor yang berperan dalam mengawasi *platform* seperti melakukan non-aktivasi pada soal-soal yang mendapat banyak laporan, serta melakukan aktivasi atau non-aktivasi pada *superadmin*. *Admin* adalah aktor yang berperan untuk membuat *course* dan soal yang dapat dikerjakan oleh *coder*. *Coder* adalah aktor yang berperan sebagai peserta yang dapat mengerjakan soal yang telah dibuat oleh *admin*.

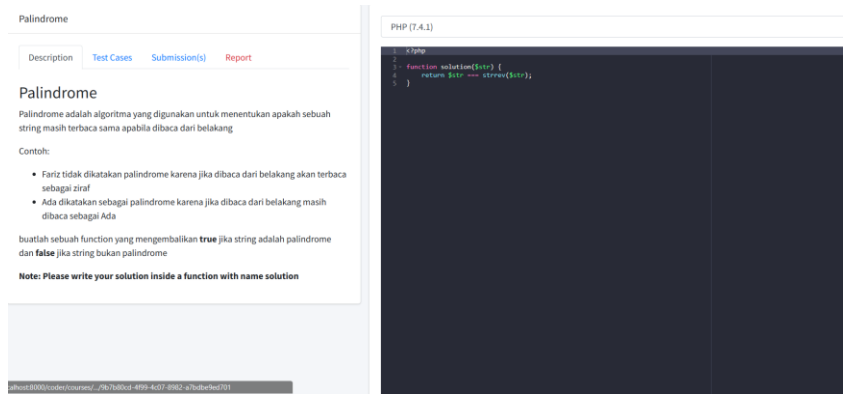


Gambar 4: Skema Relasi

Pada *platform* yang akan dibangun akan membutuhkan 10 tabel sesuai dengan Gambar 4. Data peserta *course* akan disimpan pada table *coders*, data admin yang dapat membuat *course* akan disimpan pada tabel *admins*. *course* yang dapat diikuti akan disimpan pada table *courses* dan soal-soal yang dapat dikerjakan akan tersimpan pada tabel *questions*. Setelah *coder* telah berhasil menjawab soal maka jawaban akan tersimpan pada tabel *submissions*. Daftar soal yang telah berhasil dijawab akan disimpan pada table *coder solved questions*, untuk *course* yang berhasil diselesaikan akan tersimpan pada tabel *coder complete courses*.

3.3. Coding

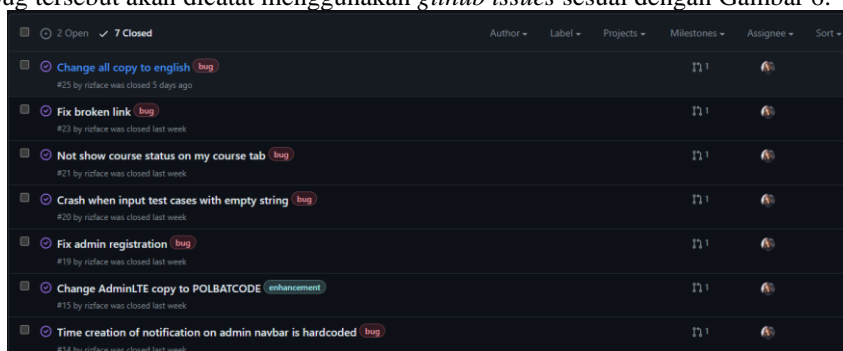
Proses *coding* dilakukan menggunakan bahasa pemrograman *PHP* dan *Laravel* sebagai *web framework*, dan menggunakan *Postgresql* sebagai basis data serta menggunakan *Ace* sebagai *embedded code editor*. *Embedded code editor* berguna untuk memberikan pengalaman IDE seperti visual studio kepada para pengguna sebagaimana ditunjukkan pada Gambar 5.



Gambar 5: Code editor

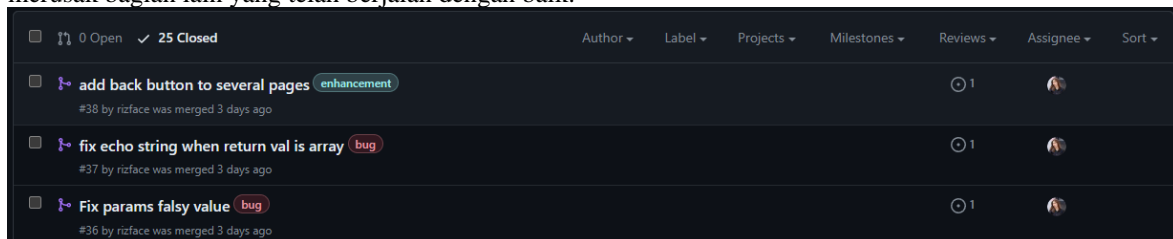
3.4. Testing

Testing pada web ini dilakukan dengan teknik *blackbox testing*. jika ditemukan *bug* dalam proses testing maka bug tersebut akan dicatat menggunakan *github issues* sesuai dengan Gambar 6.



Gambar 6: Github Issues

Setelah *bug* ditemukan maka akan dilakukan proses *bug fixing*, setelah *bug fixing* selesai maka akan dilakukan *open pull request* pada *repository* sesuai dengan Gambar 7, untuk memastikan *bug fixing* tidak merusak bagian lain yang telah berjalan dengan baik.



Gambar 7: Pull request

Testing dilakukan berdasarkan fitur-fitur yang akan dibuat sesuai dengan yang tertera pada *usecase diagram* pada Gambar 3 dan kebutuhan fungsional sesuai dengan Tabel 1. Hasil *testing* tertera pada Tabel 3.

Tabel 3: Hasil Pengujian

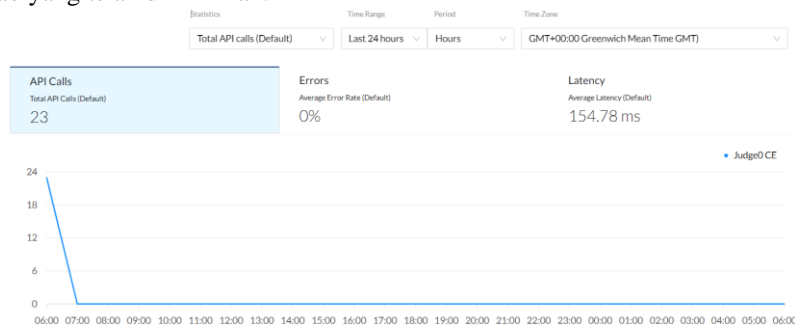
Fungsi yang diuji	Cara pengujian	Hasil yang diharapkan	Status hasil pengujian
Registrasi <i>admin course</i>	Memberikan input <i>username</i> dan <i>password</i> lalu <i>click</i> tombol <i>register</i>	<i>Admin course</i> berhasil terdaftar dan diarahkan ke halaman <i>login admin course</i>	Berhasil
Login <i>admin course</i>	Memberikan input berupa <i>username</i> dan <i>password</i> lalu <i>click</i> tombol <i>login</i>	<i>Admin course</i> berhasil login dan diarahkan ke halaman <i>dashboard</i>	Berhasil
<i>Dashboard admin course</i>	<i>Click</i> menu <i>courses</i> pada <i>navbar</i>	Menampilkan <i>dashboard</i> dan terdapat <i>tab all courses</i> dan <i>my courses</i> yang bisa menampilkan <i>courses</i>	Berhasil

Fungsi yang diuji	Cara pengujian	Hasil yang diharapkan	Status hasil pengujian
Halaman tambah <i>course</i>	<i>Click</i> tombol <i>ne course</i>	Diarahkan ke halaman untuk menambahkan <i>course</i> dan berhasil membuat <i>course</i>	Berhasil
Halaman tambah <i>question</i>	Buka halaman detail <i>course</i> lalu <i>click</i> tombol <i>new question</i>	Diarahkan ke halaman untuk menambahkan <i>question</i> dan berhasil menambahkan <i>question</i> jika semua <i>input</i> diisi dengan benar	Berhasil
Halaman edit <i>question</i>	Buka halaman detail <i>course</i> lalu <i>click</i> tombol <i>edit</i> pada <i>question</i> yang diinginkan	Diarahkan ke halaman edit <i>question</i> dan berhasil melakukan edit jika semua <i>input</i> diisi dengan benar.	Berhasil
Hapus <i>question</i>	Buka halaman detail <i>course</i> lalu <i>click</i> tombol <i>delete</i> pada <i>question</i> yang diinginkan	Berhasil menghapus <i>question</i> yang diinginkan	Berhasil
Halaman validasi <i>question</i>	Buka halaman detail <i>course</i> lalu <i>click</i> tombol <i>validate</i> pada <i>question</i> yang diinginkan	Diarahkan ke halaman validasi <i>question</i> , berhasil menuliskan <i>code</i> untuk validasi, serta berhasil melakukan pengujian otomatis	Berhasil
Registrasi <i>coder</i>	Memberikan input <i>username</i> dan <i>password</i> lalu <i>click</i> tombol <i>register</i>	<i>Coder</i> berhasil terdaftar dan diarahkan ke halaman <i>login coder</i>	Berhasil
Login <i>coder</i>	Memberikan input berupa <i>username</i> dan <i>password</i> lalu <i>click</i> tombol <i>login</i>	<i>Coder</i> berhasil login dan diarahkan ke halaman <i>dashboard</i>	Berhasil
Halaman dashboard	<i>Click</i> menu <i>courses</i> pada <i>navbar</i>	Menampilkan <i>dashboard</i> dan terdapat <i>tab all courses</i> dan <i>my courses</i> yang bisa menampilkan <i>courses</i>	Berhasil
<i>Join Course</i>	<i>Click</i> tombol <i>join</i> pada <i>course</i> yang diinginkan	<i>Coder</i> terdaftar pada <i>course</i> yang dipilih, dan dapat mengerjakan <i>question</i> yang tersedia.	Berhasil
Mengerjakan <i>question</i>	<i>Click</i> tombol <i>view</i> pada <i>question</i> yang diinginkan	<i>Coder</i> diarahkan ke halaman yang berisikan <i>code editor</i> untuk mengerjakan <i>question</i> .	Berhasil
<i>Report Question</i>	<i>Click</i> tab <i>report</i> pada halaman detail <i>question</i> dan input <i>title & description</i>	Laporan berhasil diinputkan.	Berhasil
Login <i>superadmin</i>	Memberikan input berupa <i>username</i> dan <i>password</i> lalu <i>click</i> tombol <i>login</i>	<i>Superadmin</i> berhasil login dan diarahkan ke <i>dashboard</i> .	Berhasil
Melakukan aktivasi pada <i>admin course</i>	Klik menu <i>validate admin course</i> , klik tab <i>validate</i> , dan klik tab <i>validate</i> pada salah satu <i>admin course</i>	<i>Admin course</i> dapat melakukan <i>login</i>	Berhasil
Melakukan deaktivasi pada <i>admin course</i>	Klik menu <i>validate admin course</i> , klik tab <i>invalidate</i> , dan klik <i>invalidate</i> pada salah satu <i>admin course</i>	<i>admin course</i> tidak dapat melakukan <i>login</i>	Berhasil

Melihat report yang telah diajukan untuk <i>questions</i> terkait	Klik menu <i>question reports</i> , lalu klik <i>view reports</i> pada salah satu <i>questions</i>	Sistem akan melakukan <i>redirect</i> ke halaman yang menampilkan <i>list report</i> untuk <i>questions</i> terkait	Berhasil
Melakukan deaktivasi pada <i>questions</i>	Klik menu <i>question reports</i> , lalu klik <i>invalidate question</i> pada salah satu <i>questions</i>	<i>Question</i> tidak muncul dalam <i>list question</i> .	Berhasil

3.5. Implementation

Tahap *implementation* dimulai dengan integrasi *web* yang dibangun dengan *code executor*, *code executor* yang digunakan adalah *Judge0* dengan limitasi untuk melakukan *request* sebanyak 50 *request* dalam sehari karena menggunakan versi gratis, sesuai dengan Gambar 8. *Code executor* berguna untuk melakukan eksekusi pada *code* yang telah dikirimkan.



Gambar 8: Judge0 statistic

agar *platform* ini dapat berkomunikasi dengan *Judge0* dalam proses *development*, maka perlu melakukan *deployment*, VPS yang digunakan dalam proses *development* adalah *EC2 instance* dari *amazon web services* sesuai dengan Gambar 9.



Gambar 9: AWS ec2 detail

Spesifikasi VPS yang digunakan untuk *deployment* dapat dilihat pada Tabel 4.

Tabel 4: Spesifikasi VPS

Komponen	Spesifikasi
Operating System	Ubuntu LTS 22.04
vCPU	2 Core
Storage	30 GB
Memory	1 GB
Web Server	Nginx
Database	Postgresql 16

3.6. Maintenance

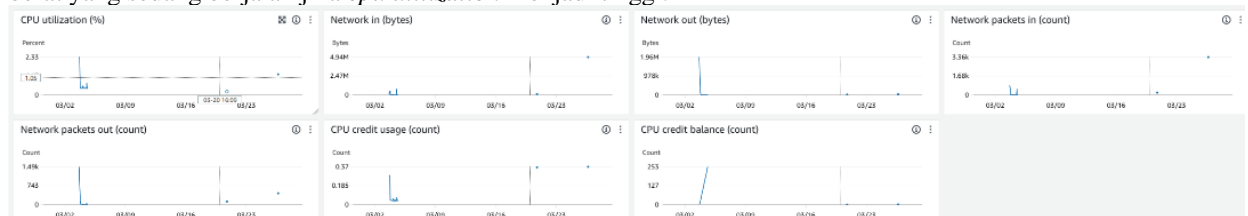
Maintenance dilakukan dengan melakukan *monitoring* pada *Laravel log* untuk mengetahui apa yang terjadi pada *web* yang sedang berjalan, sebagai contoh *log* pada Gambar 10 menunjukkan *web* gagal melakukan koneksi *database* pada 2023-03-05.

```

[2024-03-05 08:26:17] connection to server at "localhost" (127.0.0.1), port 5432 failed: Connection refused
Is the server running on that host and accepting TCP/IP connections? [Exception: "[object] PDOException(code: 7): SQLSTATE[08006] [7] connection to server at "localhost" (127.0.0.1), port 5432 failed: Connection refused
[stacktrace]
#0 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Connectors/Connector.php(65): PDO->construct()
#1 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Connectors/Connector.php(44): Illuminate\Database\Connectors\Connector->createPdoConnection()
#2 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Connectors/PostgresConnector.php(35): Illuminate\Database\Connectors\Connector->createConnection()
#3 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Connectors/ConnectionFactory.php(180): Illuminate\Database\Connectors\PostgresConnector->connect()
#4 [Internal Function]: Illuminate\Database\Connectors\ConnectionFactory->Illuminate\Database\Connectors\ConnectionFactory->close()
#5 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Connection.php(1339): call_user_func()
#6 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Concerns/ManagesTransactions.php(186): Illuminate\Database\Connection->getPdo()
#7 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Concerns/ManagesTransactions.php(152): Illuminate\Database\Connection->handleBeginTransactionException()
#8 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Concerns/ManagesTransactions.php(126): Illuminate\Database\Connection->createTransaction()
#9 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Database/Concerns/ManagesTransactions.php(24): Illuminate\Database\Connection->beginTransaction()
#10 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/DatabaseQueue.php(223): Illuminate\Database\Connection->transaction()
#11 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/Worker.php(389): Illuminate\Queue\DatabaseQueue->pop()
#12 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/Worker.php(383): Illuminate\Queue\Worker->Illuminate\Queue\Worker->[closure]()
#13 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/Worker.php(162): Illuminate\Queue\Worker->getNextJob()
#14 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/Console/WorkCommand.php(137): Illuminate\Queue\Worker->daemon()
#15 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/Console/WorkCommand.php(139): Illuminate\Queue\Console\WorkCommand->runWorker()
#16 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Queue/Console/WorkCommand.php(56): Illuminate\Queue\Console\WorkCommand->handle()
#17 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Container/Util.php(41): Illuminate\Container\BoundMethod::Illuminate\Container\{closure}()
#18 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Container/BoundMethod.php(92): Illuminate\Container\Util::unwrapClosures()
#19 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Container/BoundMethod.php(35): Illuminate\Container\BoundMethod::callBoundMethod()
#20 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Container/Container.php(662): Illuminate\Container\BoundMethod::call()
#21 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Console/Command.php(213): Illuminate\Container\Container->call()
#22 /home/ubuntu/equinix/vendor/symfony/console/Command/Command.php(326): Illuminate\Console\Command->execute()
#23 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Console/Command.php(188): Symfony\Component\Console\Command\Command->run()
#24 /home/ubuntu/equinix/vendor/symfony/console/Application.php(1096): Illuminate\Console\Command->run()
#25 /home/ubuntu/equinix/vendor/symfony/console/Application.php(324): Symfony\Component\Console\Application->doRunCommand()
#26 /home/ubuntu/equinix/vendor/symfony/console/Application.php(175): Symfony\Component\Console\Application->doRun()
#27 /home/ubuntu/equinix/vendor/laravel/framework/src/Illuminate/Foundation/Console/Kernel.php(201): Symfony\Component\Console\Application->run()
#28 /home/ubuntu/equinix/artisan(35) Illuminate\Foundation\Console\Kernel->handle()
#29 [main]
    
```

Gambar 10: Laravel log

Maintenance juga dilakukan dengan melakukan *monitoring* pada *resource* yang digunakan oleh *EC2 instance* seperti penggunaan *memory*, *storage*, dan *network* sesuai dengan Gambar 11. Monitoring ini sangat berguna untuk mengetahui jika terjadi masalah pada *platform* ini, sebagai contoh kita bisa mengetahui ada proses berat yang sedang berjalan jika *cpu utilization* menjadi tinggi.



Gambar 11: EC2 resource

Penjelasan masing – masing item pada *monitoring* berdasarkan Gambar 11 dapat dilihat pada Tabel 5

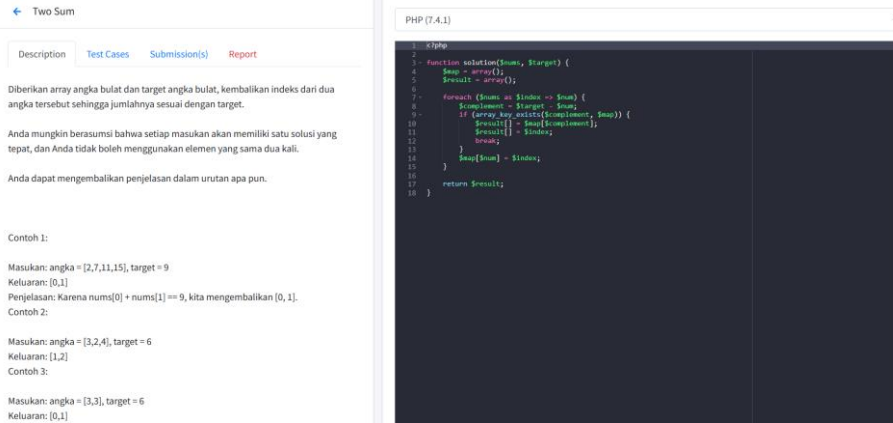
Tabel 5: Penjelasan monitoring item

Item	Penjelasan
CPU Utilization	Persentasi dari penggunaan cpu yang digunakan untuk menjalankan vps dan juga code program
Network In	Ukuran data yang diterima dari jaringan dalam satuan bytes
Network Out	Ukuran data yang dikirim keluar jaringan dalam satuan bytes
Network Packets In	Jumlah paket yang sudah diterima dari jaringan
Network Packets Out	Jumlah paket yang sudah dikirim keluar jaringan
CPU Credit Usage	Jumlah CPU booster yang digunakan untuk menambah kemampuan cpu.
CPU Credit Balance	Jumlah CPU booster yang masih tersisa dan dapat digunakan.

4. HASIL DAN PEMBAHASAN

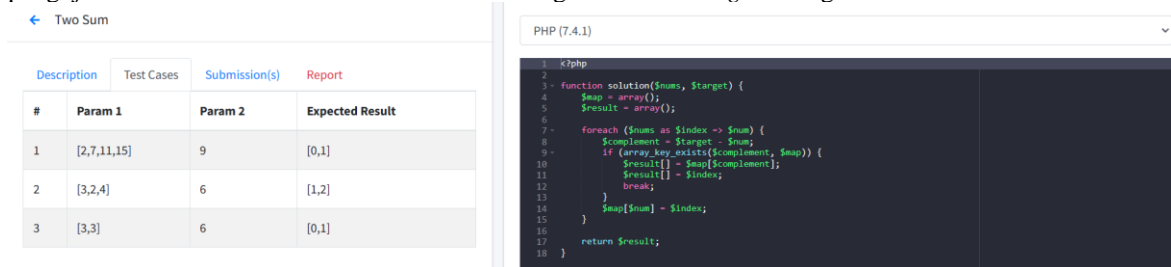
4.1. Implementasi Web

Pada Gambar 12 menampilkan halaman editor yang digunakan untuk menuliskan *code* yang akan diujikan menggunakan *testcases* yang telah tersedia. Pada tab *description* menampilkan deskripsi dan petunjuk soal yang akan dikerjakan.



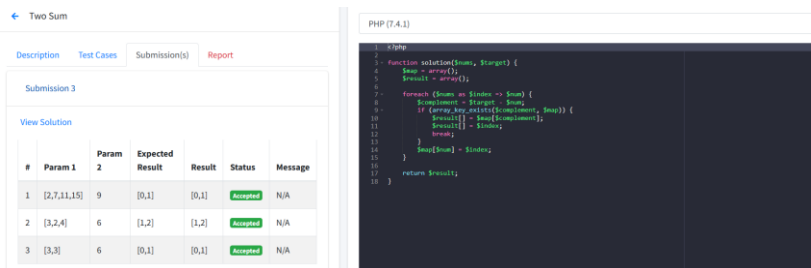
Gambar 12: Halaman Editor (Description)

Pada Gambar 13 menampilkan *testcases* yang akan digunakan untuk menguji *code* yang telah ditulis, pengujian akan dilakukan otomatis oleh sistem dengan bantuan *Judge0* sebagai *code executor*.

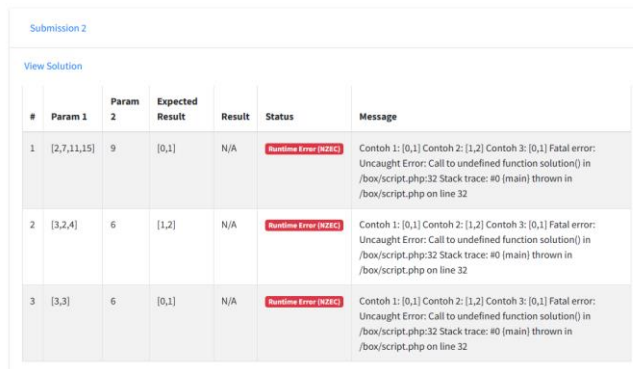


Gambar 13: Halaman Editor (Test Cases)

Pada Gambar 14 menampilkan *submission(s)* yang telah dikirimkan untuk diuji, pada tabel tersebut ditampilkan parameter yang digunakan untuk menguji *code*, lalu hasil yang diharapkan, dan juga hasil yang dihasilkan oleh *code* yang telah ditulis, jika terdapat *syntax error* dalam *code* yang ditulis maka juga akan ditampilkan pada tabel tersebut, seperti pada Gambar 15

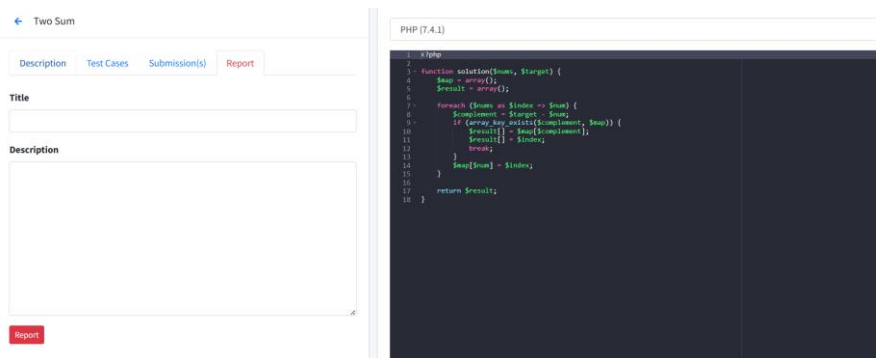


Gambar 14: Submissions Success



Gambar 15: Submissions Syntax Error

Pada Gambar 16 menampilkan tab *report* yang bisa digunakan untuk mengirimkan laporan jika dirasa ada yang salah dengan soal sehingga soal tidak dapat dikerjakan.



Gambar 16: Report Question

5. KESIMPULAN & SARAN

5.1 Kesimpulan

Berdasarkan pembahasan sebelumnya, maka dapat diambil kesimpulan bahwa *Platform Competitive Programming* Sebagai Media Belajar Struktur Data Berbasis Web telah berhasil dibangun menggunakan metode *waterfall* dalam waktu 5 bulan. *Platform* ini dapat digunakan untuk membuat dan mengerjakan soal struktur data dilengkapi dengan *code editor* dan pengujian *code* secara otomatis yang dapat meningkatkan pengalaman belajar struktur data yang diharapkan dapat memperkuat pemahaman fundamental tentang struktur data dan algoritma. Saran untuk penelitian selanjutnya adalah menambahkan bahasa dukungan untuk berbagai bahasa pemrograman dalam penyelesaian soal, serta memasukan *time complexity* dan *space complexity* dalam penilaian jawaban.

DAFTAR PUSTAKA

- [1] Nair, P. R. (2020). Increasing employability of Indian engineering graduates through experiential learning programs and competitive programming: Case study. *Procedia Computer Science*, 172, 831-837.
- [2] Arief, A., Muhammad, M., & Amin, F. (2023). Pengembangan Game Edukasi dengan Metode GDLC: Studi Kasus Mata Kuliah Algoritma dan Struktur Data. *IJIS-Indonesian Journal on Information System*, 8(2), 120-125.
- [3] Nita, S., & Pratiwi, A. O. C. (2022). Implementasi Augmented Reality (AR) sebagai Sarana Media Pembelajaran Interaktif Algoritma Pemrograman. *Jurnal Bina Komputer*, 4(1), 1-8.
- [4] Damopolii, Z. D. P., Koniyo, M. H., & Takdir, R. (2021). Perancangan Media Pembelajaran Berbasis Audio Visual Pada Materi Looping Mata Kuliah Algoritma Dan Pemograman Dasar Universitas Negeri Gorontalo. *Inverted: Journal of Information Technology Education*, 1(2), 64-74
- [5] Katahman, M. A., & Othman, M. F. (2021). Pembangunan Aplikasi Realiti Terimbuah Untuk Pengenalan Struktur Data. *Applied Information Technology and Computer Science*, 2(2), 364-377.
- [6] Khairunnisa, K., & Wulan, N. (2020). Perancangan Intelligent Tutoring System Sebagai Upaya Inovatif Pada Pembelajaran Algoritma dan Struktur Data. *Algoritma. J. Ilmu Komput. Dan Inform*, 4(2).
- [7] Mambang, S. P. C., Marleny, F. D., Ansari, N. H., Baddrudin, A., Yenitia, A., Dixky, M., ... & Salsabila12, T. (2022). E-Padi Berbasis Android Untuk Meningkatkan Minat Generasi Muda Pada Sektor Pertanian. *Jurnal Nasional Komputasi dan Teknologi Informasi*, 5(1).
- [8] Purnamasari, D. A., Uperiati, A., Rokhayati, Y., Alif Perdana, I. N., & Karisa, V. (2023). Rancang Bangun Aplikasi Pemandu Wisata dengan Penerapan Google API & PWA. *Rancang Bangun Aplikasi Pemandu Wisata Dengan Penerapan Google API & PWA*. <https://doi.org/10.25047/jtit.v10i2.332>
- [9] Setiyani, L. (2021, November). Desain Sistem: Use Case Diagram. In *Prosiding Seminar Nasional Inovasi dan Adopsi Teknologi (INOTEK)* (Vol. 1, No. 1, pp. 246-260).
- [10] Dalimunthe, Abdul Haris. "Desain Basis Data Sistem Pemesanan Makanan dan Minuman Pada Restoran Berbasis Online." *Jurnal Rekayasa Elektro Sriwijaya* 1.2 (2020): 53-61.
- [11] Himanen, E. (2023). Developing a feature to a CRM system with Laravel