


SISTEM INFORMASI DAN UNDIAN PERTANDINGAN BADMINTON MENGGUNAKAN METODE SISTEM KEJUARAAN PBSI

Zidan Dhiyaul Haq¹, Aris Rakhmadi²

¹Universitas Muhammadiyah Surakarta, Jl. A. Yani, Pablean, Kartasura, Sukoharjo,
Indonesia

²Universitas Muhammadiyah Surakarta, Jl. A. Yani, Pablean, Kartasura, Sukoharjo,
Indonesia

 zidan.zd49@gmail.com

Abstrak. Banyaknya peraih medali para atlet nasional Indonesia menjadikan badminton sebagai salah satu cabang olahraga paling populer di Indonesia. Masyarakat berharap prestasi ini dapat terus berlanjut dan semakin berkembang. Antusias masyarakat Indonesia dapat dilihat dari banyaknya pelatihan dan kompetisi cabang olahraga di tingkat SMP, SMA, maupun Universitas mulai dari tingkatan lokal sebagai lomba tujuhbelasan, kabupaten, maupun nasional. Namun pengelolaan kompetisi tersebut kebanyakan masih menggunakan sistem konvensional. Penelitian ini mencoba menawarkan alternatif sistem informasi untuk mengelola sebuah turnamen kompetisi dengan lebih praktis dan transparan dengan menerapkan aturan kompetisi dari PBSI dan *Badminton World Federation*. Sistem dikembangkan dengan bahasa pemrograman PHP dan *framework* Laravel 12 untuk mencapai performa terbaik dengan database MySQL dan Laragon sebagai *platform* pengembangan serta Rumahweb sebagai penyedia domain hosting dengan nama domain *turneygen.online*. Pengujian sistem dilakukan dengan metode *blackbox testing* dan *System Usability Scale* dari 25 responden. Seluruh fitur berhasil melewati proses uji *blackbox* dengan hasil yang sesuai dengan harapan peneliti. Aplikasi berhasil membuat bagan pertandingan sistem eliminasi tunggal yang sesuai dengan aturan BWF dalam penempatan *byes* dan *seed*. Hasil akhir penelitian menunjukkan bahwa web aplikasi TurneyGen mendapat skor 78 dalam uji SUS dan termasuk dalam rentang C(70-80) yang dinilai 'Baik' dan 'Dapat diterima' sehingga dapat dikatakan sistem layak digunakan dan memenuhi ekspektasi pengguna.

Kata kunci: sistem informasi, pertandingan, PBSI, badminton, undian



PENDAHULUAN

Banyak bidang olahraga yang menjadi favorit warga Indonesia, salah satunya adalah badminton. Kepopuleran olahraga badminton tidak dipungkiri karena banyaknya prestasi yang diraih para atlet Indonesia di kompetisi internasional, seperti *Thomas Cup*, *Uber Cup*, maupun kompetisi yang digelar di bawah naungan BWF (*Badminton World Federation*) lainnya. Indonesia juga merupakan salah satu negara dengan minat tertinggi pada olahraga badminton menurut laporan Kementerian Pemuda dan Olahraga [1]. Kepopuleran badminton ini tentunya diikuti dengan banyaknya kompetisi atau turnamen lokal yang diadakan untuk terus menumbuhkan bibit-bibit unggul atlet untuk generasi selanjutnya.

Kompetisi atau turnamen memiliki standar dari organisasi pengelola bidang yang dipertandingkan. Di Indonesia, PBSI (Persatuan Bulutangkis Seluruh Indonesia) merupakan organisasi yang menjadi pusat segala aturan dan ketentuan dalam bermain badminton, maka dari itu kompetisi badminton yang diadakan di Indonesia juga harus mengikuti ketentuan dari PBSI. Banyak pertimbangan yang harus dilakukan dalam proses *drawing* atau pengundian, diantaranya adalah efektifitas dan keabsahan hasil pengundian oleh panitia. Pengundian secara manual biasanya dilakukan dengan mengambil kertas bertuliskan nama peserta lalu menempatkan nama yang muncul tadi ke bagan pertandingan. Cara ini dinilai tidak efisien karena memerlukan banyaknya prasarana, waktu, dan mengharuskan adanya saksi mata. Selain itu adanya potensi kecurangan dalam proses pengundian juga menjadi masalah di sini. Salah satu bentuk kecurangan dalam kompetisi olahraga adalah *matchfixing* atau bisa disebut dengan pengaturan skor [2] Manipulasi skor bisa berjalan dengan mulus apabila pihak lawan bisa diajak bekerjasama. Pencegahan bisa dilakukan dengan metode pengacakan yang baik, sehingga peserta tidak saling bertemu peserta dari latar belakang yang sama.

Dalam kompetisi badminton, sistem gugur merupakan format yang paling umum digunakan. Sistem gugur ini membuat setiap peserta bermain sekali pertandingan dan akan langsung tereliminasi pada kekalahan pertama. Lawan yang akan dihadapi ditentukan oleh panitia melalui proses yang disebut *draw*. Menurut [3], ada beberapa aturan mengenai cara menentukan bagan pertandingan dalam proses *draw* atau pengundian : (1) Ketika jumlah peserta berjumlah 4, 8, 16, 32, 64, 128 ataupun semua pangkat kelipatan 2 selanjutnya, mereka akan mendapatkan masing-masing 1 lawan pas. (2) Ketika jumlah peserta bukan merupakan pangkat kelipatan 2, maka harus ada *by* pada



ronde pertama. Jumlah *by* harus sama dengan selisih antara jumlah peserta dan pangkat kelipatan 2 di atasnya (contoh, jika jumlah peserta adalah 17 maka $32 - 17 = 15$ merupakan jumlah *bye*). Ada ketentuan pemberian *seed* dalam setiap kompetisi. Menurut aturan dari BWF, 2 *seed* merupakan jumlah maksimal untuk 16 peserta, 4 *seed* untuk 16 - 31 peserta, 8 *seed* untuk 32 - 63 peserta, 16 untuk 64 - 128 peserta, dan 32 *seed* untuk 129 atau lebih peserta [3]. Masih menurut aturan dari BWF, peserta dengan *seed* 1 dan 2 akan selalu terpisah yakni *seed* 1 di paling atas dan *seed* 2 di paling bawah. Selain *seed* 1 dan 2, ada juga aturan untuk peletakan *seed* yang diambil dari *General Competition Regulations Part III-Section 1A General Competition Regulations PART III SECTION 1 A GENERAL COMPETITION REGULATIONS 1. PURPOSE AND APPLICABILITY OF THE REGULATIONS 1.1* BWF pada Tabel 1 sebagai contoh.

Tabel 1. *Placing of byes and seeds in a draw* [4]

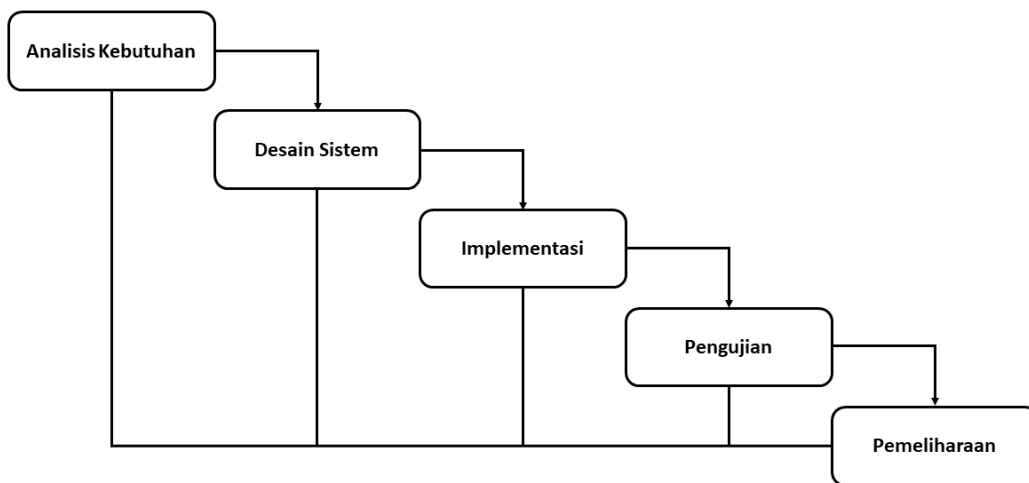
Entries	No of seeds	Seed 1	Seed 2	Seeds 3/4	Seeds 5/8	No of byes	Placement of byes
17	4	1	32	9,24	-	15	2, 4, 6, 8, 10, 12, 14, 16, 19, 21, 23, 25, 27, 29, 31
18	4	1	32	9,24	-	14	2, 4, 6, 8, 10, 12, 14, 19, 21, 23, 25, 27, 29, 31
19	4	1	32	9,24	-	13	2, 4, 6, 8, 10, 12, 14, 19, 21, 23, 27, 29, 31
20	4	1	32	9,24	-	12	2, 4, 6, 10, 12, 14, 19, 21, 23, 27, 29, 31
21	4	1	32	9,24	-	11	2, 4, 6, 10, 12, 14, 19, 23, 27, 29, 31
22	4	1	32	9,24	-	10	2, 4, 6, 10, 14, 19, 23, 27, 29, 31
23	4	1	32	9,24	-	9	2, 4, 6, 10, 14, 19, 23, 27, 31
24	4	1	32	9,24	-	8	2, 6, 10, 14, 19, 23, 27, 31
25	4	1	32	9,24	-	7	2, 6, 10, 14, 23, 27, 31
26	4	1	32	9,24	-	6	2, 6, 10, 23, 27, 31
27	4	1	32	9,24	-	5	2, 6, 10, 23, 31
28	4	1	32	9,24	-	4	2, 10, 23, 31
29	4	1	32	9,24	-	3	2, 10, 31
30	4	1	32	9,24	-	2	2, 31
31	4	1	32	9,24	-	1	2,
32	8	1	32	9,24	5, 13, 20, 28	0	-



Semua proses di atas tentu akan memakan banyak sekali waktu, mengingat peserta suatu kompetisi tidak selalu pasti jumlahnya. Pengelola kompetisi lokal yang tidak memiliki banyak tenaga kerja pasti merasa aturan dari BWF tidaklah efisien sehingga kebanyakan penyelenggara akan mengabaikan Sebagian besar aturan tersebut. Maka dari itu peneliti berkeinginan untuk membuat alternatif alat untuk membuat bagan secara otomatis tanpa harus mengabaikan aturan dari BWF ataupun PBSI. Beberapa hal yang ingin diwujudkan peneliti dalam sistem informasi ini adalah registrasi peserta, penentuan kemampuan atau unggulan peserta, proses *draw* atau pengundian, *input* dan *monitoring* pertandingan, dan rekam jejak pertandingan turnamen.

METODE

Tahap pengembangan aplikasi akan menggunakan metode pengembangan *waterfall*. *Waterfall* adalah salah satu model SDLC yang dalam alurnya dimulai dari tahap analisis kebutuhan urut sampai dengan tahap pemeliharaan [5]. Tahapan proses pengembangan aplikasi dengan metode *waterfall* adalah seperti pada Gambar 1.



Gambar 1. Diagram SDLC Waterfall

1. Perancangan Sistem

a. Analisis Kebutuhan

Tahap analisis kebutuhan merupakan langkah pertama dalam proses pengembangan perangkat lunak. Peneliti menganalisis kebutuhan perangkat lunak dengan memanfaatkan lembar aturan kompetisi dari BWF. Dengan beberapa pertimbangan, dapat diambil kesimpulan bahwa sistem akan memiliki 3 jenis pengguna, yaitu *user* admin, panitia turnamen dan peserta.



User panitia turnamen dapat mengakses beberapa fitur tambahan seperti membuat turnamen, mengelola turnamen, mengelola bagan pertandingan, menambah peserta, dan mengelola postingan. Pada sisi *user* peserta dapat mencari turnamen, mendaftar turnamen, dan mengelola profilnya. Sedangkan admin bisa melihat seluruh data *user*. Beberapa uraian di atas dapat diambil kesimpulan bahwa kebutuhan fungsional dari perangkat lunak adalah sebagai berikut:

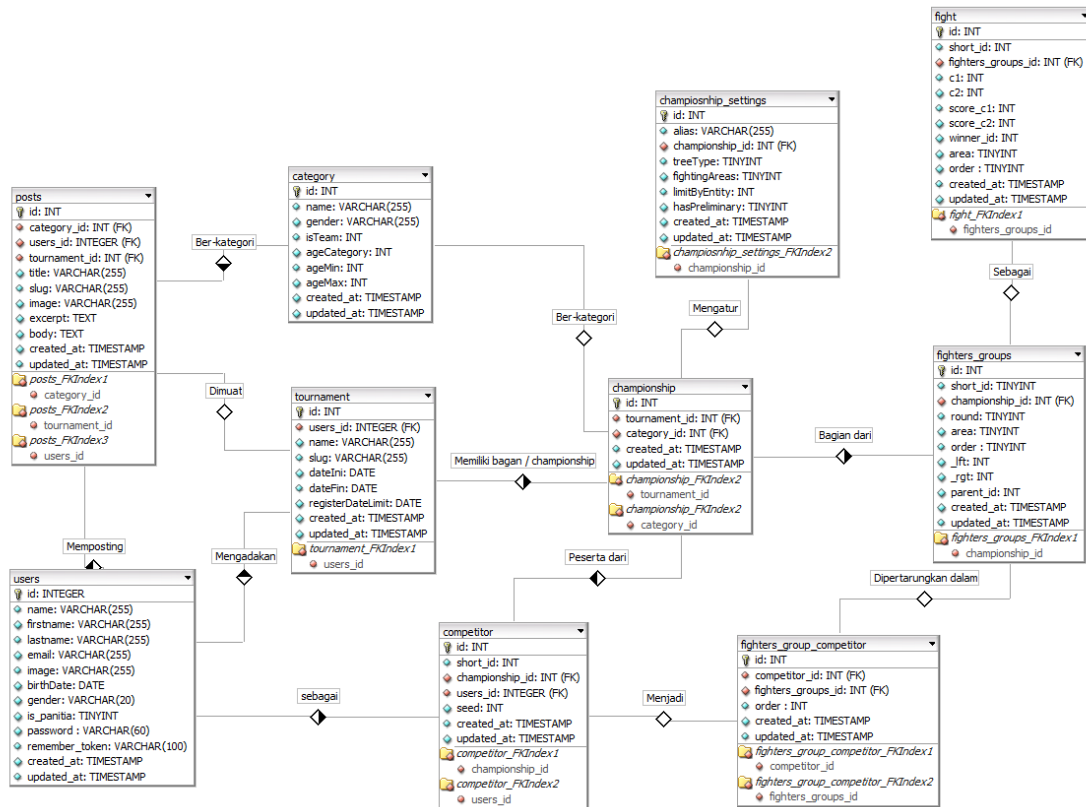
1. Sistem memiliki menu login sebagai syarat untuk mengakses fitur utama.
2. Sistem memiliki *dashboard* untuk membuat turnamen, mengelola turnamen, mengatur peserta dan pertandingan, dan membuat postingan turnamen.
3. Sistem memiliki halaman admin untuk mengelola data *user*.
4. Setiap turnamen yang sudah selesai tetap akan disimpan dan bisa dilihat kembali sebagai dokumentasi.

Selain beberapa hal di atas, ada juga kebutuhan non-fungsional untuk pengembangan perangkat lunak. Sistem informasi ini nantinya akan membutuhkan internet untuk diakses. Pada proses perancangan akan digunakan Visual Studio Code sebagai IDE pemrograman utama dengan Github sebagai *version control*. Bahasa yang digunakan adalah PHP dengan *framework* Laravel 12 dan database MySQL. Pada sisi perangkat keras, proses pembuatan aplikasi akan menggunakan Laptop Asus Vivobook X415EP yang ditenagai *processor* generasi ke-11 dari Intel yakni Intel i5-1135G7 dengan penyimpanan SSD 512GB + HDD 1TB, dan GPU NVIDIA GeForce MX330.

b. Desain Sistem

Tahapan setelah analisis kebutuhan adalah merancang desain sistem. Perancangan dilakukan dengan diagram *Unified Modeling Language* (UML) yang terdiri dari *use case diagram* dan *activity diagram*. Sebelum itu peneliti mempersiapkan desain database yang akan digunakan oleh sistem. Pada sistem informasi turnamen badminton ini terdapat 10 entitas yakni *user*, *tournament*, *posts*, *category*, *championship*, *championship_settings*, *competitor*, *fight*, *fighters_groups*, dan *fighters_group_competitor*. Berikut adalah desain database dari sistem dalam bentuk *Entity Relationship Diagram*.

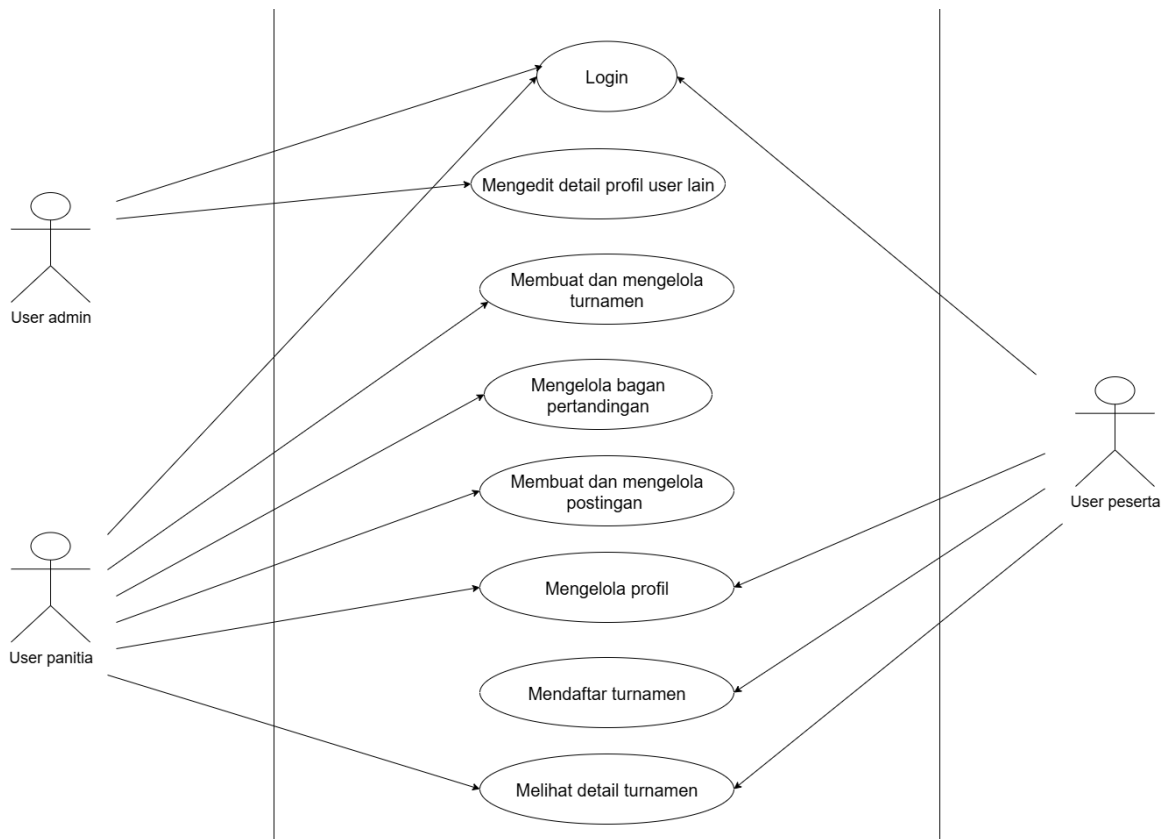




Gambar 2. Desain database

Use case diagram bertujuan memberikan visual dari interaksi sistem yang mungkin terjadi dengan user [6]. Sistem generator badminton memiliki 3 aktor yang akan mengoperasikannya. Aktor pertama yakni user admin, lalu ada user panitia, dan yang ketiga adalah user peserta turnamen. User yang teridentifikasi atau mendaftar sebagai panitia akan memiliki dashboard untuk mengelola sebuah turnamen. User panitia dapat membuat turnamen, lalu memilih turnamen yang ingin dibuka dan dikelola. Setiap panitia dapat membuat banyak turnamen sekaligus. Sedangkan untuk user biasa atau peserta akan ke halaman pencarian turnamen. Peserta dapat mendaftar ke sebuah turnamen setelah melengkapi semua data pribadi pada halaman profil. Lalu yang terakhir adalah user admin. Admin hanya bisa melihat data seluruh user namun hanya bisa mengganti password user lain. Gambar 3 di bawah ini merupakan use case diagram dari sistem informasi turnamen badminton.



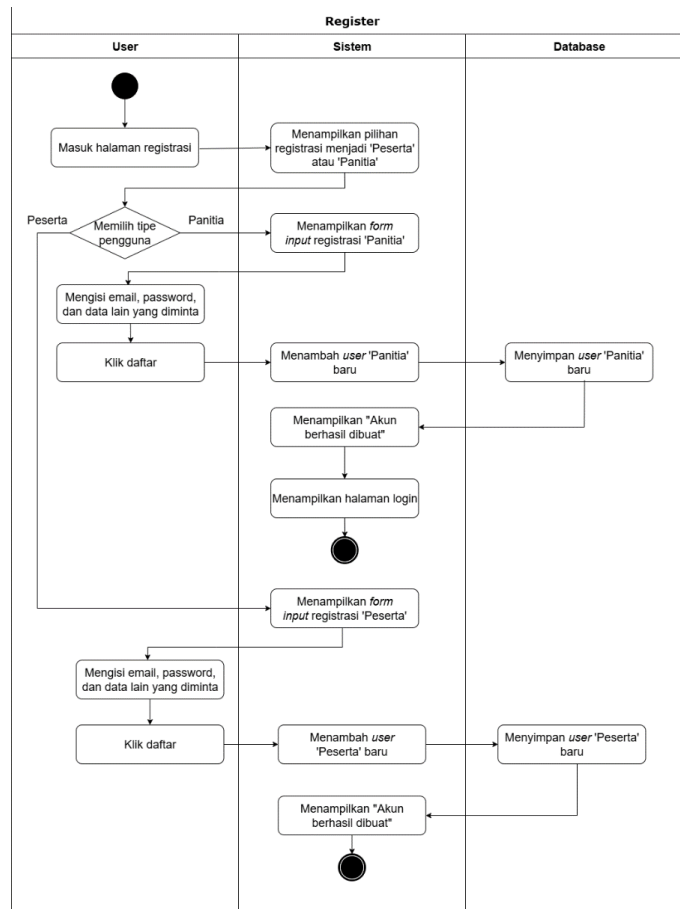


Gambar 3. Use Case Diagram sistem

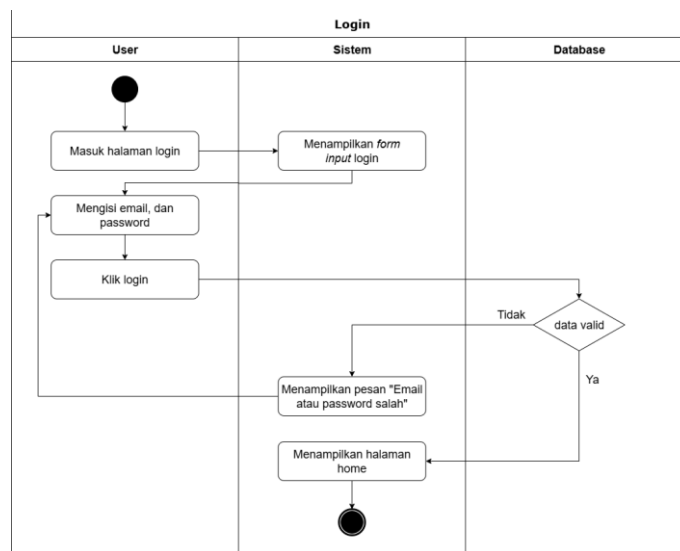
UML activity diagram (UAD) merupakan notasi grafis yang digunakan untuk menjelaskan perilaku dari *instance class* UML selama sebuah sistem informasi digunakan. Di dalamnya digunakan *Object Constraint Language* (OCL) untuk membatasi dan menjelaskan setiap tindakan yang dilakukan oleh masing-masing *instance class*. [7].

Gambar 4 menjelaskan alur proses ketika seseorang ingin melakukan pendaftaran sebagai pengguna. Pengguna akan diminta mengisi email, kata sandi, dan beberapa data diri seperti yang tercantum pada database untuk melakukan pendaftaran. Setelah itu pengguna baru bisa melakukan *login* dengan akun yang telah dibuat tadi seperti pada Gambar 5.





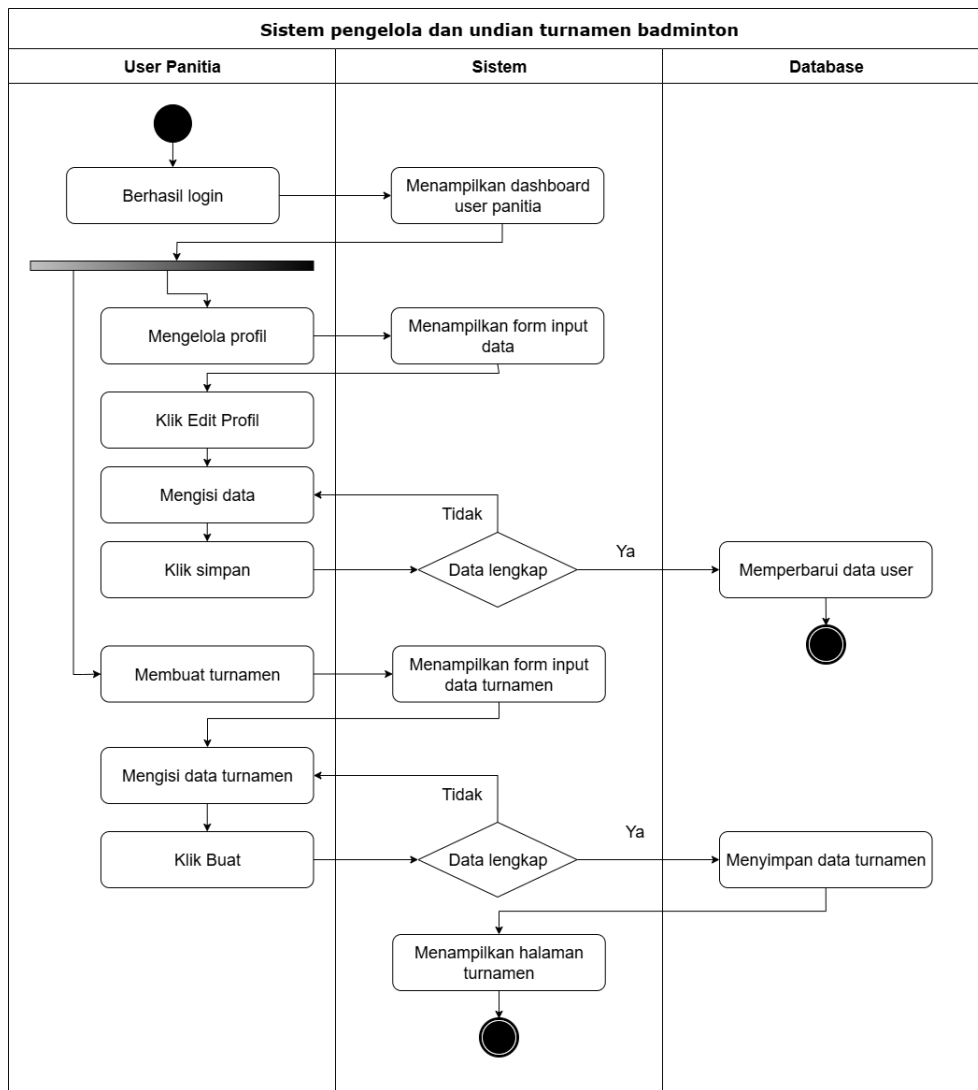
Gambar 4. Diagram Aktifitas proses Register



Gambar 5. Diagram Aktifitas proses Login

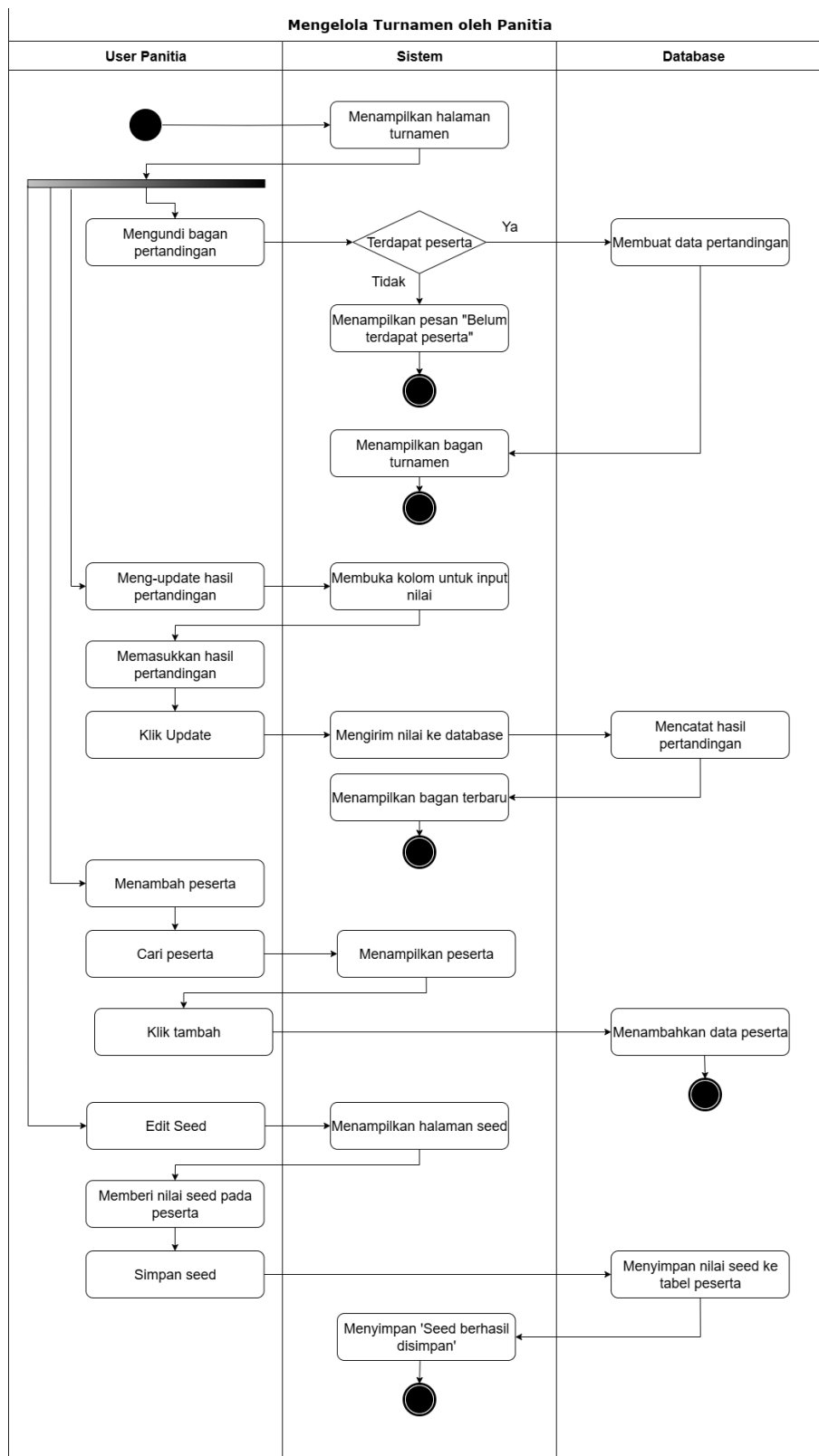


Pengelola turnamen akan memiliki halaman awal login yang berbeda dengan peserta. Alur kegiatan pengelola turnamen seperti membuat turnamen, mengelola turnamen, mengundi bagan pertandingan, maupun membuat postingan dapat dilihat melalui diagram aktifitas pada Gambar 6. Untuk alur mengelola turnamen bisa dilihat pada Gambar 7. Setelah membuat turnamen, panitia juga bisa membuat postingan agar peserta bisa mendaftar. Alur untuk mengelola postingan ada pada Gambar 8.



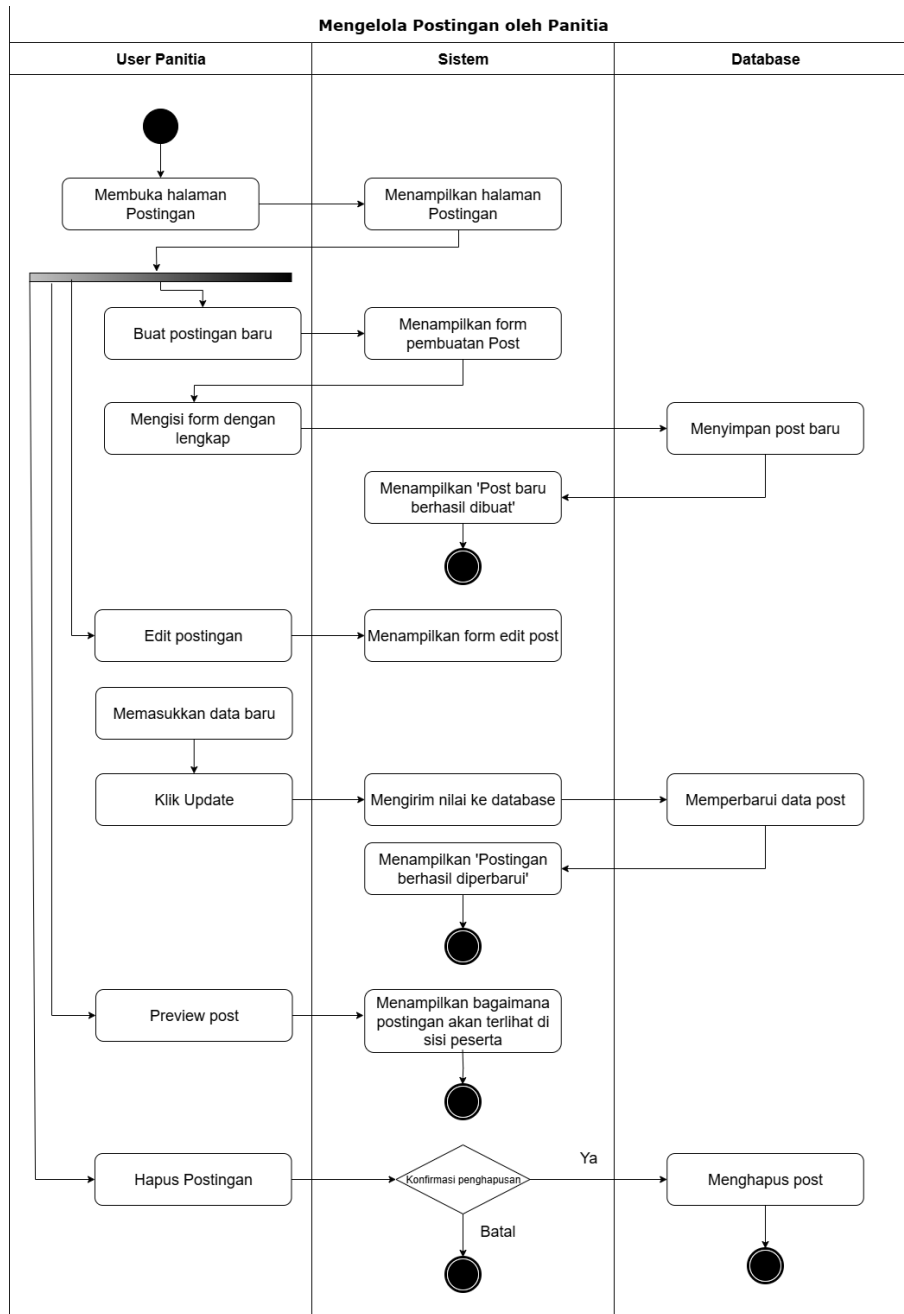
Gambar 6. Diagram Aktifitas pada *dashboard* Panitia





Gambar 7. Diagram Aktifitas mengelola turnamen oleh Panitia

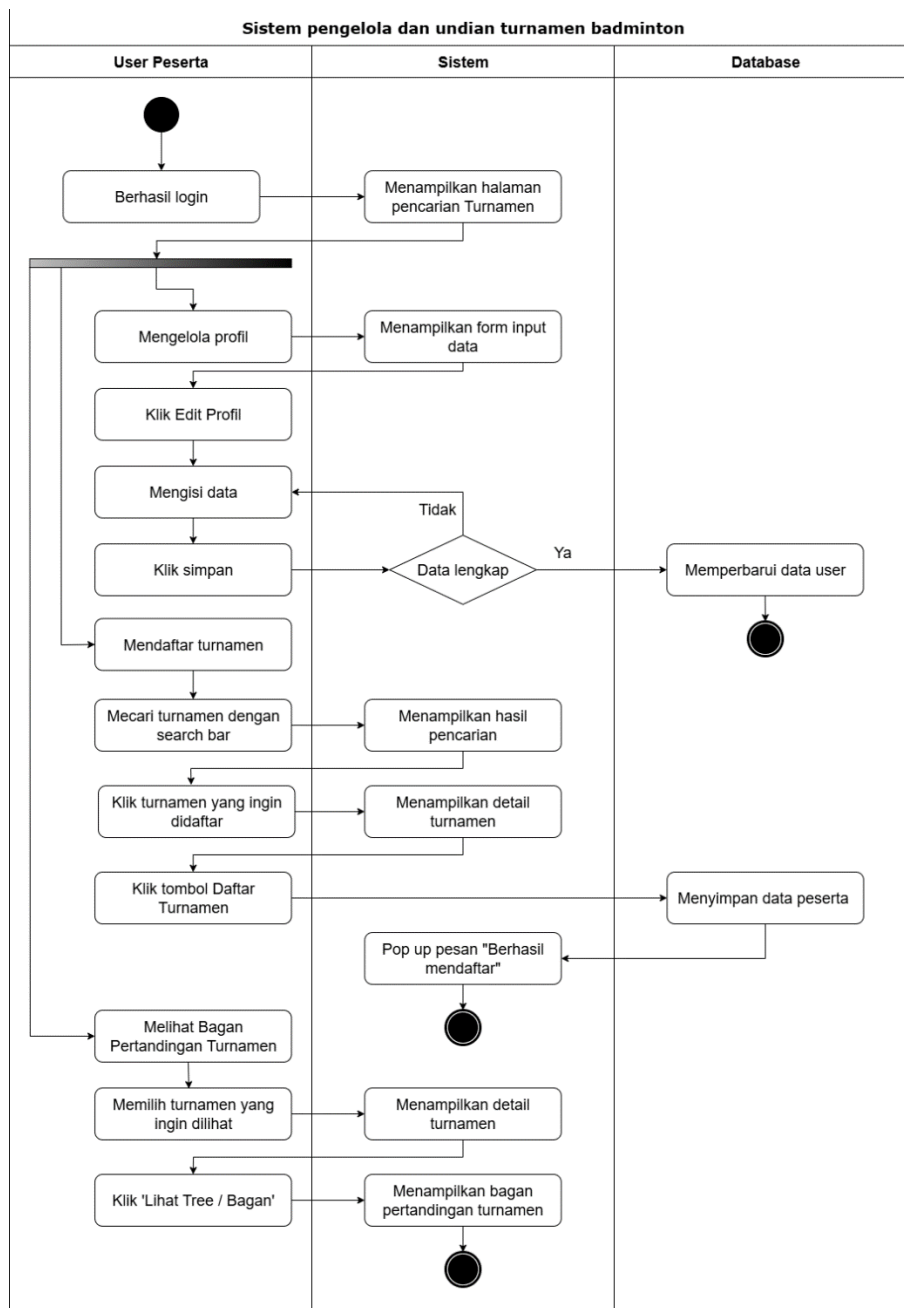




Gambar 8. Diagram Aktifitas mengelola postingan oleh Panitia

Halaman pertama yang dibuka peserta setelah *login* adalah halaman pencarian turnamen. Peserta dapat mencari turnamen, melihat deskripsi turnamen, dan mendaftar turnamen dengan alur seperti pada Gambar 9.

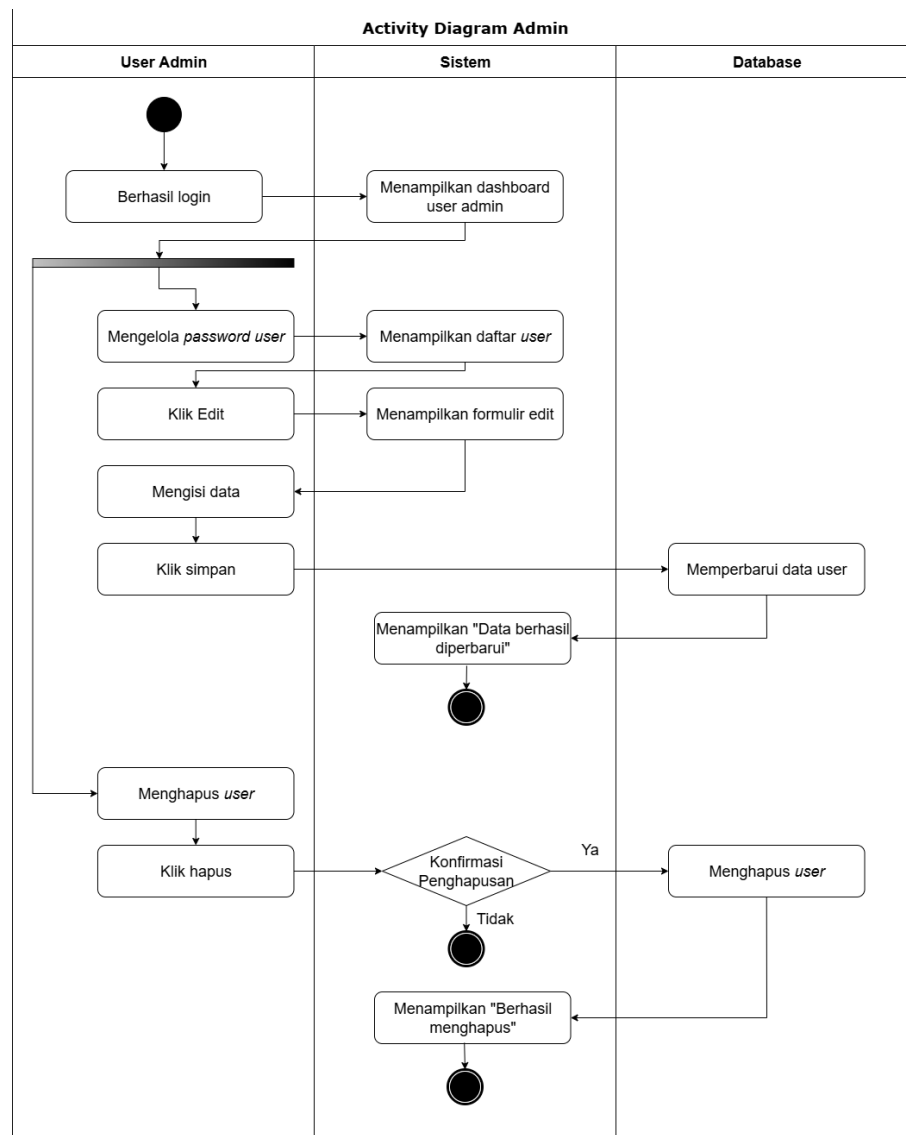




Gambar 9. Diagram Aktifitas Peserta

Terakhir adalah diagram aktifitas admin yang dapat dilihat pada Gambar 10. Admin bisa melihat data semua *user* dalam tabel, namun hanya bisa mengedit *password* saja. Adanya *user* admin di sini sebagai antisipasi jika pengguna melupakan data akun yang sudah dibuat maka bisa menghubungi admin untuk meminta detail dari akun si pengguna tadi.

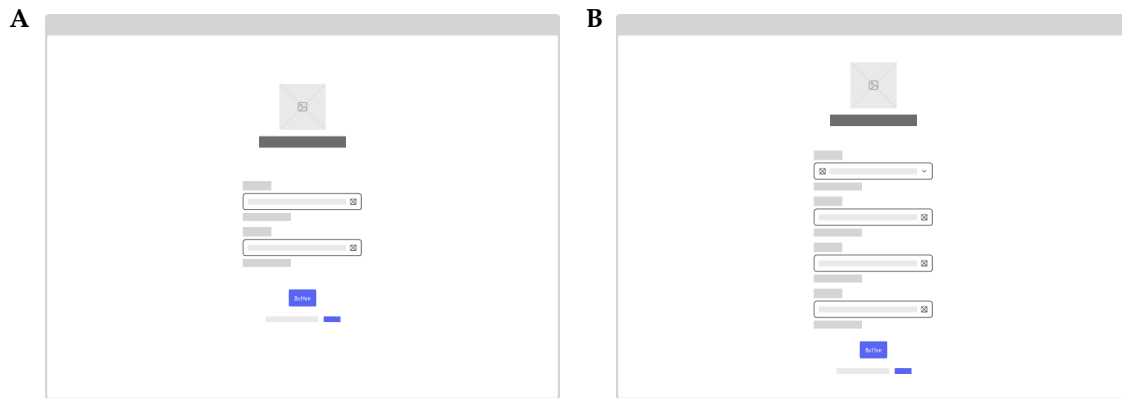




Gambar 10. Diagram Aktifitas Admin

Desain antarmuka dibuat secara *low fidelity* untuk menggambarkan tata letak konten yang akan dilihat pengguna ketika menggunakan sistem informasi undian badminton. Halaman pertama yang akan dilihat pengguna adalah halaman *login* yang tampak pada Gambar 11 (A). Jika pengguna belum memiliki akun, pengguna bisa menekan *link* dibawah tombol *login* dan akan langsung diarahkan ke halaman *register* dengan tampilan seperti pada Gambar 11 (B). Pengguna juga akan menentukan tipe akun yang akan dibuat, akun panitia atau akun peserta.





Gambar 11. Desain *low fidelity* (A) Halaman login, dan (B) Halaman registrasi

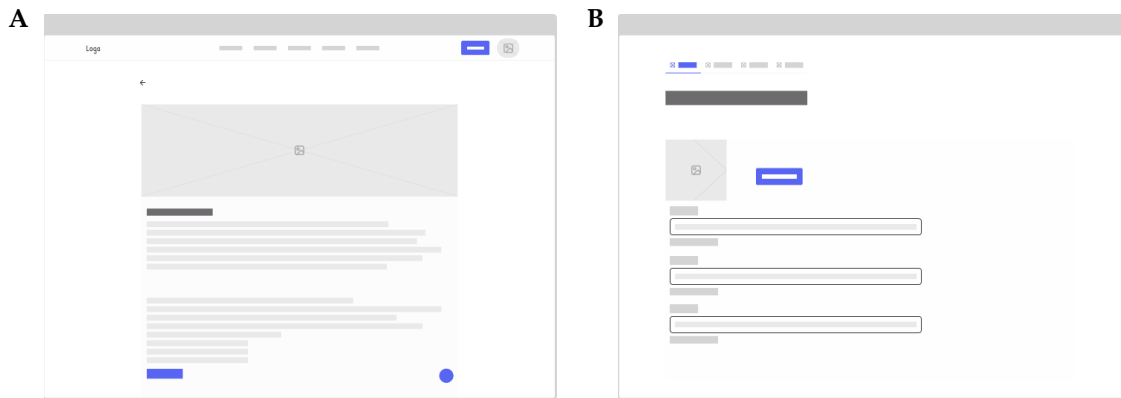
Setelah berhasil login, maka pengguna akan diarahkan ke halaman *dashboard*. Pengguna panitia dan pengguna peserta akan memiliki *dashboard* yang berbeda. Pada Gambar 12 (A) diperlihatkan panitia akan memiliki bilah menu disamping yang mana ada beberapa tab fitur yang dapat digunakan seperti membuat turnamen, dan mengelola turnamen. Sedangkan peserta akan langsung diarahkan pada halaman pencarian turnamen seperti pada Gambar 12 (B). Setelah menemukan turnamen yang dicari, peserta dapat melihat detail turnamen dan mendaftar.



Gambar 12. Desain *low fidelity 2* (A) Halaman *dashboard*, dan (B) Halaman pencarian turnamen

Gambar 13(A) bawah ini merupakan gambaran sederhana bagaimana tata letak konten pada halaman detail turnamen. Fungsi *edit* profil dapat diakses oleh semua pengguna. Pengguna akan diarahkan ke halaman profil dengan tampilan seperti pada Gambar 13(B).





Gambar 13. Desain *low fidelity* 3 (A) Halaman detail turnamen, dan (B) Halaman profil

2. Implementasi

Implementasi merupakan tahapan setelah membuat desain dari sebuah sistem informasi. Website dibuat menggunakan Bahasa pemrograman PHP dengan *framework* Laravel 12. Data dari sistem akan disimpan dalam database MySQL menggunakan *Database Management System* Laragon. Setelah web selesai dikerjakan, selanjutnya akan dideploy ke domain hosting dari Rumahweb dengan nama domain *turneygen.online*.

3. Pengujian

Langkah selanjutnya setelah sistem berhasil dibangun adalah proses pengujian supaya sistem berjalan sesuai dengan yang diharapkan [8]. Penelitian ini akan menggunakan uji *blackbox* untuk mengevaluasi sistem sesuai kebutuhan pengguna dengan memperhatikan *input* dan *output* dari sistem. [9] Metode ini tidak mengharuskan penguji mengetahui kode internal dari aplikasi, sehingga metode ini juga biasa disebut dengan pengujian berbasis spesifikasi dan pengujian perilaku [10] Sistem juga akan diuji menggunakan *System Usability Scale* (SUS). SUS adalah skala ukuran untuk menilai kegunaan suatu sistem. Sebuah kuisisioner singkat dengan sepuluh pertanyaan yang di setiap pertanyaannya menggunakan skala Likert mulai dari sangat setuju, hingga sangat tidak setuju. Tujuan rancangan ini dibuat adalah sebagai sarana penilaian dengan skor berkisar mulai 0 sampai 100. [11] Target pengguna adalah seorang pemain bulutangkis. Kuesioner akan disebar melalui media sosial WhatsApp dan distribusi langsung kepada pemain bulutangkis di Gedung Badminton Tanujayan yang berlokasi di RT01 RW01 Dusun Tanujayan, Desa Gawan, Kecamatan Colomadu Kabupaten Karanganyar.



4. Pemeliharaan

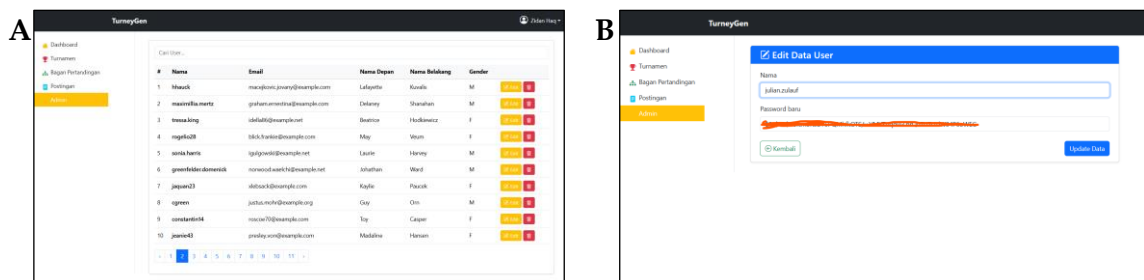
Pemeliharaan setelah sistem berhasil dikembangan sangat diperlukan untuk mencegah virus atau program berbahaya lainnya masuk ke dalam sistem [8] Patch aplikasi akan dirilis apabila muncul beberapa masalah dalam aplikasi. Selain itu, patch bisa juga digunakan untuk meningkatkan kualitas dari sistem tanpa harus menunggu masalah terjadi. [12]

HASIL

1. Web Aplikasi TurneyGen

Web bernama TurneyGen yang terhosting di *domain* turneygen.online adalah hasil dari penelitian ini. Pengguna bisa langsung mengakses link *domain* untuk menggunakannya. Diharapkan web aplikasi ini bisa menjadi alternatif pengelola turnamen dan media distribusi untuk menyebarluaskan sebuah turnamen kepada calon peserta yang diinginkan oleh penyelenggara.

TurneyGen memiliki 3 jenis pengguna, yakni admin, peserta dan panitia. Pengguna admin bisa melihat data pengguna namun hanya bisa mengubah *password* sebagai Batasan. Penampakan halaman admin bisa dilihat pada Gambar 14.



Gambar 14. Halaman admin (A) Halaman daftar *user*, dan (B) Halaman edit *password*

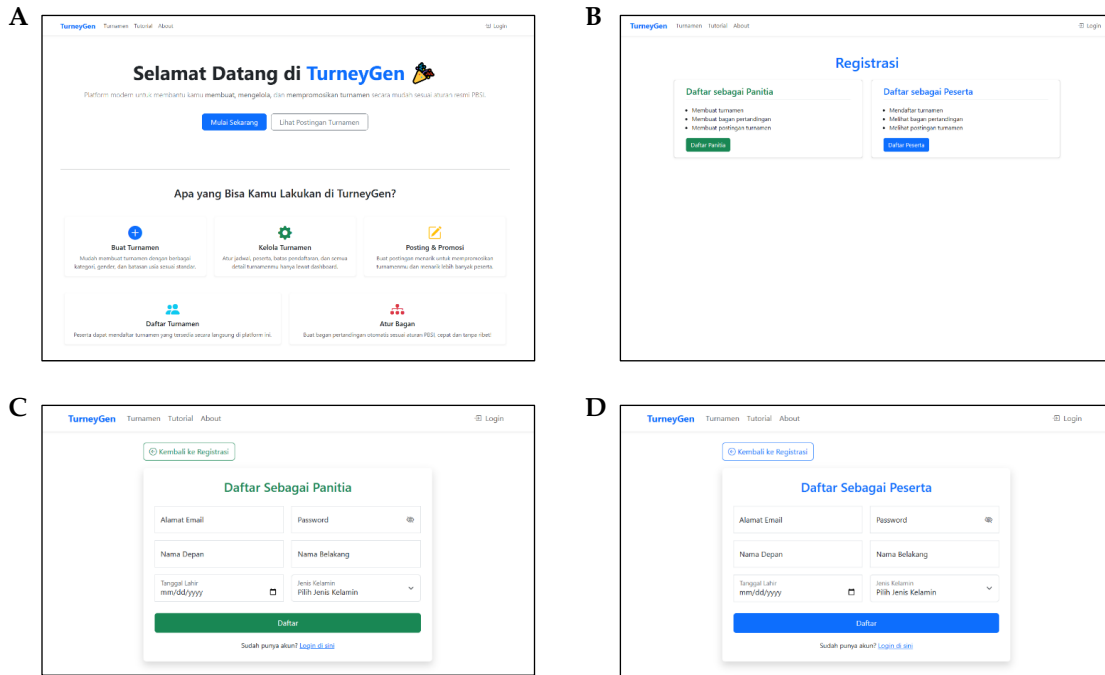
Pengguna panitia bisa menggunakan fitur utama dari aplikasi ini yaitu mengelola turnamen, dan mengelola postingan untuk turnamen, sedangkan pengguna peserta hanya bisa sebatas melihat detail dan mendaftar turnamen yang sudah diunggah oleh para pengguna panitia.

Saat pertama kali mengakses turneygen.online pengguna akan disambut dengan sebuah landing page yang menjelaskan deskripsi tentang aplikasi ini seperti pada Gambar 15(A). Selanjutnya pengguna bisa menggunakan aplikasi sebagai *guest* atau langsung mendaftar untuk login.

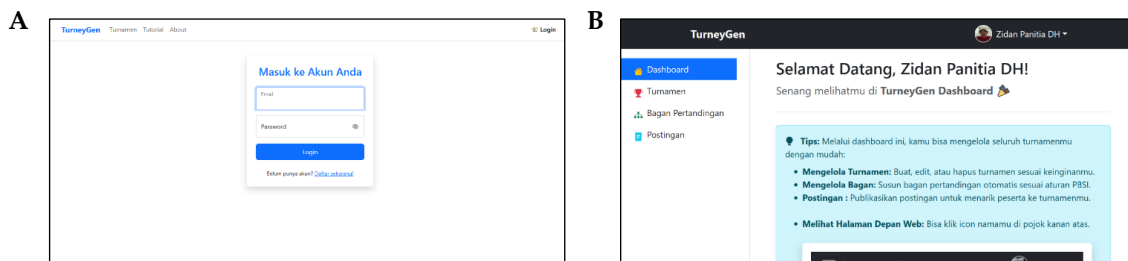
Agar bisa login pengguna harus mendaftar terlebih dahulu, yang mana mereka bisa mendaftar sebagai sebagai panitia atau peserta. Setelah memilih, pengguna akan diarahkan pada formulir pendaftaran.



Setelah berhasil mendaftar maka pengguna bisa login dengan akun yang telah didaftarkan lewat halaman login seperti pada Gambar 16(A). Selanjutnya diarahkan ke *dashboard*, di mana *user* panitia akan langsung diarahkan saat login seperti pada Gambar 16(B).



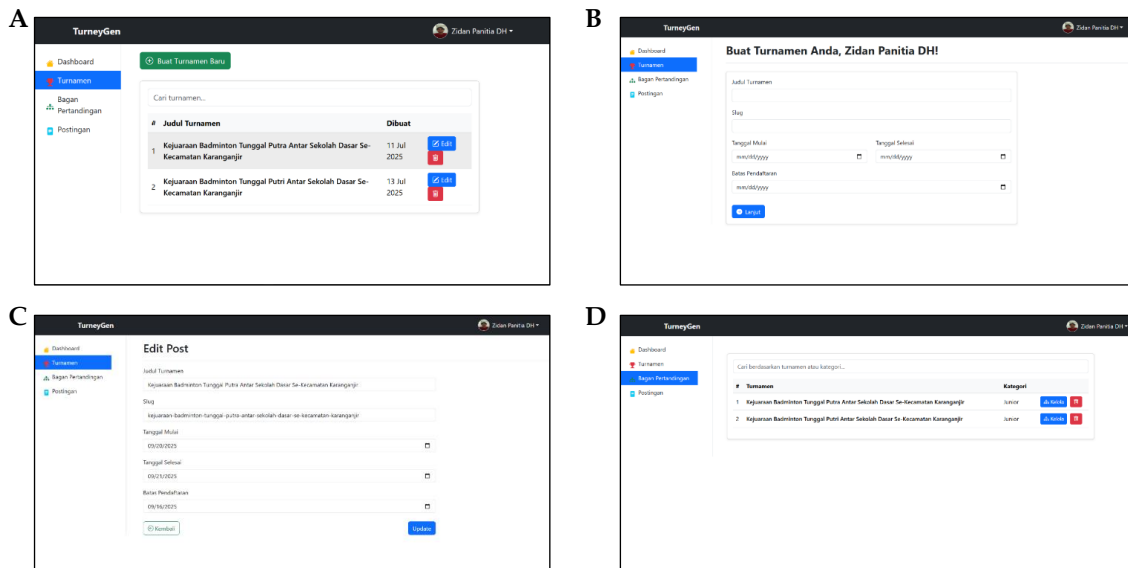
Gambar 15. Landing Page (A) Halaman home, (B) Halaman registrasi, (C) Formulir registrasi panitia, dan (D) Formulir registrasi peserta



Gambar 16. Halaman login (A) Login, dan (B) Halaman dashboard ketika berhasil login sebagai panitia



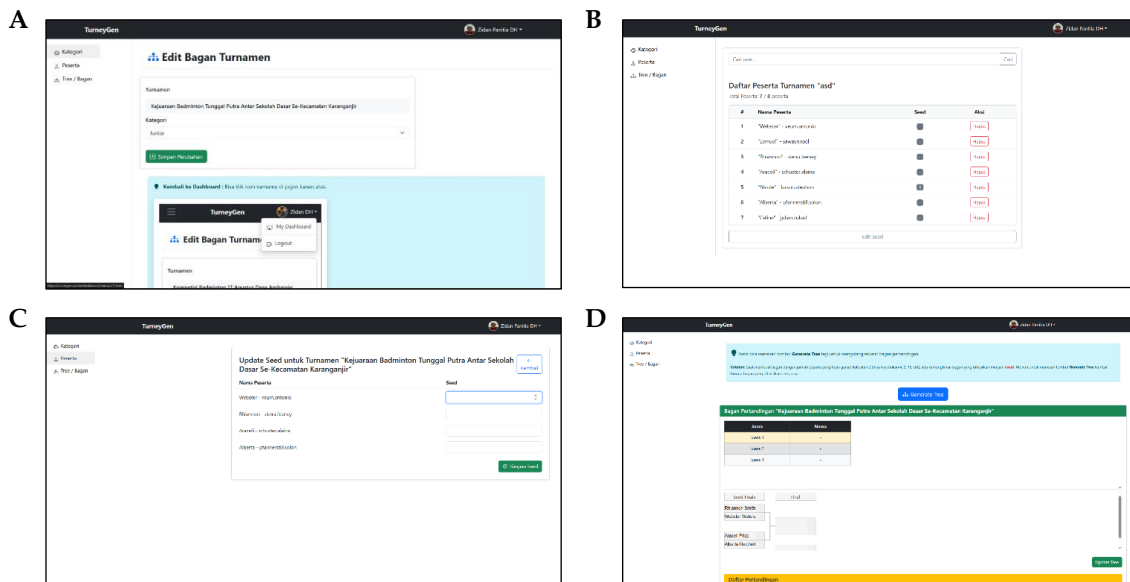
Di *dashboard* turnamen panitia bisa membuat turnamen dengan klik 'Buat Turnamen Baru' seperti pada Gambar 17(A) yang nantinya akan diarahkan ke form pembuatan turnamen. Setelah berhasil membuat turnamen *user* akan diarahkan ke halaman bagan. Jika *user* ingin mengubah data turnamen yang telah dibuat, terdapat halaman untuk mengubahnya. Tekan 'Edit' pada turnamen yang ingin diubah dan halaman *form edit* akan muncul.



Gambar 17. *Dashboard* turnamen (A) *List* turnamen, (B) Buat turnamen baru, (C) Edit turnamen, dan (D) Halaman kelola bagan

Dengan menekan 'Kelola' bagan yang diinginkan, *user* akan diarahkan pada halaman seperti Gambar 18(A). Di sana *user* bisa mengubah kategori turnamen. Halaman untuk mengelola peserta terlihat dan panitia bisa mengakses halaman *seed*. Setelah seluruh peserta lengkap, panitia bisa memulai turnamen dengan membuat bagannya di halaman bagan/*tree*. Pada gambar terlihat tombol 'Generate Tree' untuk membuat bagan, tabel daftar pemenang turnamen, dan bagan pertandingannya. Tombol *update* digunakan jika *user* ingin memasukkan skor hasil pertandingan, dan pemenang akan langsung maju ke babak selanjutnya.

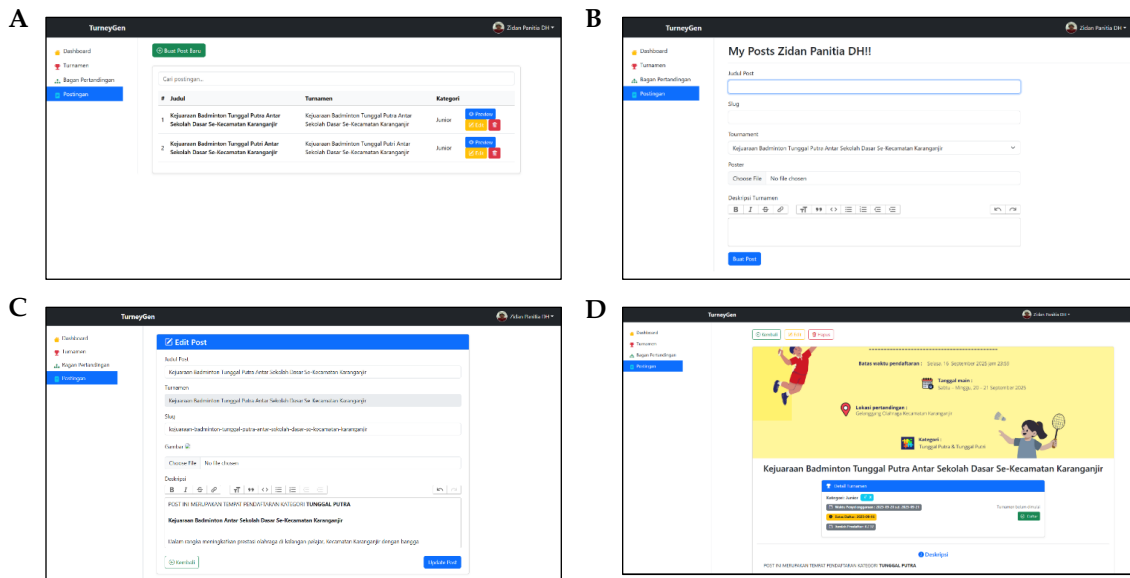




Gambar 18. Halaman kelola bagan (A) Mengganti kategori, (B) Mengelola peserta, (C) Mengelola seed peserta, dan (D) Halaman bagan pertandingan

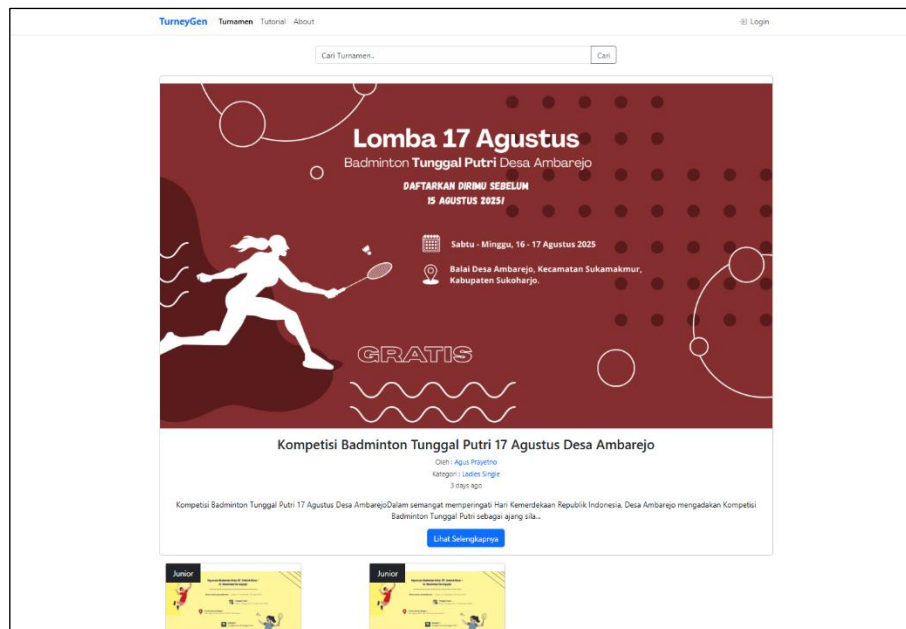
Kembali ke *dashboard*, terdapat satu bagian lagi yaitu halaman mengelola postingan. Pada Gambar 19(A) terlihat daftar post yang sudah dibuat panitia. Post dibuat berdasarkan turnamen yang telah dibuat, terlihat form memilih turnamen untuk dimuat dalam post. Post yang sudah dibuat masih bisa diubah lagi isinya dengan klik 'Edit', dan halaman *edit* akan muncul. Panitia juga bisa melihat sudut pandang peserta pada halaman *preview*. Gambar 19(D) merupakan halaman *preview* yang isinya sama dengan penampakan detail postingan di halaman depan.





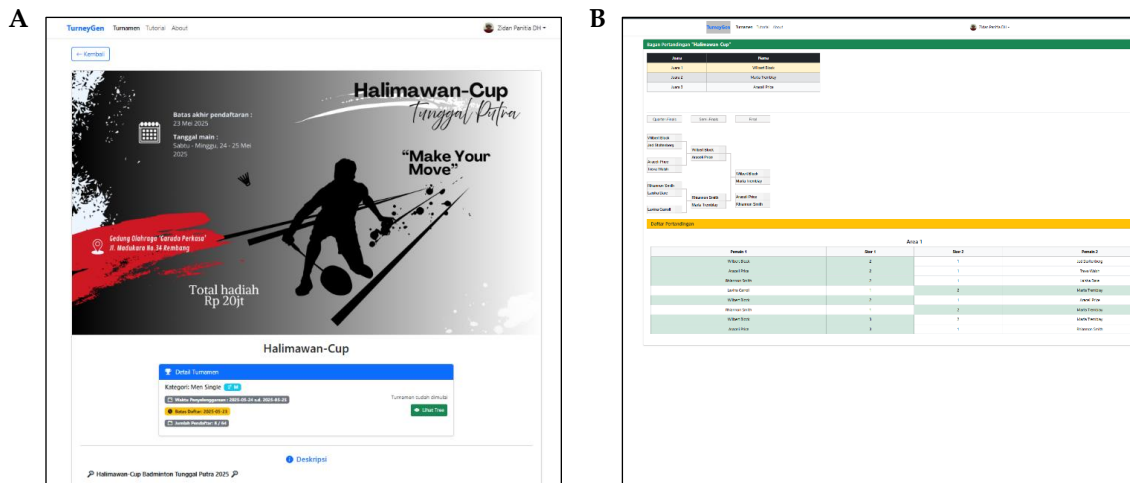
Gambar 19. Dashboard postingan (A) List postingan, (B) Buat postingan baru, (C) Edit postingan, dan (D) Halaman preview postingan

Terakhir, halaman dimana *user* peserta akan diarahkan setelah login yakni halaman pencarian turnamen akan terlihat seperti Gambar 20. Barulah jika melihat detail turnamen halaman seperti Gambar 21(A) muncul. Peserta bisa membaca detail, mendaftar turnamen, dan melihat bagan pertandingan jika memang turnamen sudah dimulai.



Gambar 20. Halaman pencarian turnamen





Gambar 21. Halaman detail turnamen (A) Detail dan deskripsi turnamen, dan (B) Bagan pertandingan turnamen

2. Uji Blackbox

Uji *blackbox* dapat dilakukan tanpa harus memiliki pengetahuan pemrograman mendalam [13]. Fokus utama pengujian ini adalah hasil dan *output* yang sesuai harapan pengembang [14]. Seluruh fitur yang dibutuhkan pengguna dapat berfungsi dengan baik. Ada beberapa penanganan *error* yang disisipkan dalam aplikasi seperti mencegah pengguna memasukkan angka pada kolom nama saat mendaftar. *Input* skor pertandingan menggunakan selain angka juga sudah diantisipasi oleh pengembang. Berikut hasil uji *blackbox* disajikan pada Tabel 2.



Tabel 2. Uji *blackbox*

No	Komponen yang diuji	Skenario Pengujian	Hasil yang diharapkan	Kesesuaian Hasil
1.	Registrasi	Membuat akun baru dengan registrasi	Pengguna berhasil mendaftar	Sesuai
2.	Login	Login menggunakan akun yang sudah dibuat	Berhasil login	Sesuai
3.	Logout	Lupa <i>password</i> Menekan tombol logout	Mengirim <i>link</i> reset <i>password</i> Berhasil Logout	Sesuai
4.	Halaman <i>Dashboard</i>	Mengakses halaman <i>dashboard</i> tanpa login atau dengan pengguna peserta	Halaman tidak bisa diakses oleh pengguna tamu dan peserta	Sesuai
5.	Menu Turnamen	Membuat, mengedit, dan menghapus turnamen	Pengguna dapat membuat, mengedit, dan menghapus turnamen	Sesuai
6.	Kelola Bagan	Menambah peserta secara manual, mengganti kategori, dan membuat bagan turnamen	Pengguna bisa menambah peserta secara manual, mengubah kategori, dan membuat bagan turnamen	Sesuai
7.	Kelola Kategori Turnamen	Mengubah kategori turnamen sesuai keinginan	Kategori turnamen berubah sesuai input	Sesuai
8.	Menu Post	Membuat, mengedit, menghapus, dan melihat <i>preview</i> postingan	Pengguna bisa membuat, mengedit, menghapus, dan melihat <i>preview</i> postingan	Sesuai
9.	Halaman Pencarian turnamen	Mencari turnamen dengan kolom pencarian	Menampilkan hasil pencarian sesuai input	Sesuai
10.	Halaman detail turnamen	Melihat deskripsi dan bagan turnamen Mendaftar turnamen	Menampilkan detail, dan bagan jika turnamen sudah dimulai Berhasil mendaftar turnamen jika memenuhi syarat	Sesuai



PEMBAHASAN

1. Sistem Undian Pertandingan

Undian peserta dalam proses pembuatan bagan pertandingan otomatis adalah fokus utama dalam penelitian ini. Beberapa persamaan matematika digunakan demi mencapai bagan yang sesuai dengan aturan PBSI. Dari penempatan peserta unggulan atau *seed*, penempatan *byes*, bahkan pada metode pengacakan peserta harus diperhatikan.

Langkah pertama, jumlah slot harus ditentukan sebagai patokan dalam proses pembuatan bagan. Ketika peserta berjumlah 4, 8, 16, 32 atau semua angka kelipatan 2 selanjutnya, setiap peserta harus mendapat satu lawan. Sedangkan untuk jumlah pemain tidak tepat angka kelipatan 2, maka harus ada *bye* pada ronde pertama pertandingan. Semua ini diatur dalam *BWF Statues, section 5.3.8* [3] Diperlukan sebuah fungsi matematika untuk menentukannya dalam aplikasi TurneyGen dimuat dalam (1).

$$s = 2^{\lceil \log_2(\max(1,a)) \rceil} \quad (1)$$

dengan :

a = total peserta turnamen

s = jumlah slot yang akan dibuat bagan

$\max(1,a)$ digunakan untuk mencegah $\log(0)$.

Contoh kasus : Turnamen Badminton Desa Wonorejo memiliki 14 peserta, maka

$$a = 14$$

$$s = 2^{\lceil \log_2(\max(1,14)) \rceil}$$

$$s = 2^{\lceil \log_2(14) \rceil}$$

$$\log_2(14)$$

$$\log_2(2 * 7)$$

$$1 + \log_2(7)$$

$$1 + 2,8074 = 3,8074$$

$$s = 2^{\lceil 3,8074 \rceil}$$

$$s = 2^4$$

$$s = 16$$

Jadi untuk jumlah peserta sebanyak 14 orang, akan menggunakan basis bagan 16 pemain.

Persamaan tadi diimplementasikan dalam fungsi *adjustFightersGroupWithByes()* seperti pada Gambar 22(A) di bawah ini.



A

```

public function adjustFightersGroupWithByes($fighters, $fighterGroups): Collection
{
    $tmpFighterGroups = clone $fighterGroups;

    // Hitung jumlah fighter sesudah repart
    $max = $this->getMaxFightersByEntity($tmpFighterGroups);
    $fighters = $this->repart($fighterGroups, $max);

    // Optionally shuffle if not in test
    if ($app()->runningUnitTests()) {
        $fighters = $fighters->shuffle();
    }

    // Hitung slot kelipatan 2 terdekat
    $total = $fighters->count();
    $slots = pow(2, ceil(log(max(1, $total), 2)));

    // Hitung jumlah bye
    $numBye = $slots - $total;

    // Gunakan insertByes yang benar
    $fighters = $this->insertByes($fighters, $numBye);

    return $fighters;
}
                
```

B

```

private function generateTraditionalSeedingOrder($slots)
{
    if ($slots == 2) return [1, 2];

    $order = [1, 2];
    while (count($order) < $slots) {
        $next = [];
        $max = count($order) * 2 + 1;
        foreach ($order as $n) {
            $next[] = $n;
            $next[] = $max - $n;
        }
        $order = $next;
    }
    return $order;
}
                
```

Gambar 22. Implementasi persamaan (A) Fungsi menghitung *byes*, dan (B) Fungsi mengurutkan peserta dengan *seed*

Setelah jumlah slot untuk basis bagan ditentukan selanjutnya adalah menentukan urutan peserta. Penelitian ini bertujuan membuat sebuah sistem undian otomatis yang sesuai dengan aturan PBSI, yang mana aturan undian tersebut menggunakan cara undian tradisional. Gambar 22(B) adalah implementasi undian sesuai aturan PBSI dalam aplikasi TurneyGen. Fungsi ini dimulai dengan statement jika slot = 2, maka langsung kembalikan array \$order = [1, 2]. Lalu dilanjutkan iterasi dengan kondisi bahwa jumlah anggota array \$order kurang dari nilai \$slots. Setelah urutan ditentukan, barulah penempatan *bye* dilakukan. Penempatan *bye* juga memiliki aturan, yang mana peserta unggulan akan mendapat prioritas untuk mendapat *bye* pada ronde pertama. Gambar 23 adalah implementasi dari penempatan *bye* yang cocok untuk aplikasi ini. Contoh iterasi untuk fungsi yang termuat dalam Gambar 22(B) ada dibawah ini.

Dimulai dengan: [1, 2]

Iterasi 1 (4 slot):

- $max = 2 * 2 + 1 = 5$
- Untuk setiap n:
 - 1 → tambahkan 1, $5 - 1 = 4$
 - 2 → tambahkan 2, $5 - 2 = 3$
- Jadi: [1, 4, 2, 3]

Iterasi 2 (8 slot):

- $max = 4 * 2 + 1 = 9$
- Loop setiap elemen dari [1, 4, 2, 3]:
 - 1 → 1, 8
 - 4 → 4, 5
 - 2 → 2, 7
 - 3 → 3, 6
- Hasil: [1, 8, 4, 5, 2, 7, 3, 6]

Iterasi 3 (16 slot):

- $Max = 8 * 2 + 1 = 17$
- Loop setiap elemen dari [1, 8, 4, 5, 2, 7, 3, 6]:
 - 1 → 1, 16
 - 8 → 8, 9
 - 4 → 4, 13
 - 5 → 5, 12
 - 2 → 2, 15
 - 7 → 7, 10
 - 3 → 3, 14
 - 6 → 6, 11
- Hasil: [1, 16, 8, 9, 4, 13, 5, 12, 2, 15, 7, 10, 3, 14, 6, 11]



```

*/
private function insertByes(Collection $fighters, int $numByeTotal)
{
    $slots = pow(2, ceil(log(max(1, $fighters->count() + $numByeTotal), 2)));
    $seedOrder = $this->generateTraditionalSeedingOrder($slots);
    $filledSlots = [];

    $seeded = $fighters->filter(fn($f) => $f->seed !== null)->sortBy('seed')->values();
    $unseeded = $fighters->filter(fn($f) => $f->seed === null)->values();
    $unseeded = $this->shuffleWithLCG($unseeded);
    // $unseeded = $fighters->filter(fn($f) => $f->seed === null)->shuffle()->values();

    $criticalSlots = [];
    foreach ($seeded as $index => $fighter) {
        $seedSlot = $seedOrder[$index];
        $filledSlots[$seedSlot] = $fighter;

        // slots untuk lawan seeded
        $matchIndex = intdiv($seedSlot - 1, 2);
        $slot1 = $matchIndex * 2 + 1;
        $slot2 = $matchIndex * 2 + 2;
        $opponentSlot = ($seedSlot === $slot1) ? $slot2 : $slot1;

        $criticalSlots[] = $opponentSlot;
    }

    $bye = $this->createByeFighter();
    foreach ($criticalSlots as $slot) {
        if ($numByeTotal <= 0) {
            break;
        }

        if (!isset($filledSlots[$slot])) {
            $filledSlots[$slot] = $bye;
            $numByeTotal--;
        }
    }

    foreach ($seedOrder as $slot) {
        if (!isset($filledSlots[$slot])) {
            if ($unseeded->isNotEmpty()) {
                $filledSlots[$slot] = $unseeded->shift();
            } elseif ($numByeTotal > 0) {
                $filledSlots[$slot] = $bye;
                $numByeTotal--;
            } else {
                $filledSlots[$slot] = $bye;
            }
        }
    }

    return collect($filledSlots)->sortKeys()->values();
}

```

Gambar 23. Fungsi *insertBye()*

Fungsi di atas mengurutkan dahulu peserta sesuai nomor unggulan atau *seed* dengan memanggil fungsi sebelumnya pada Gambar 22(B). Selanjutnya akan dibuat *bye* sebagai *placeholder* untuk dimasukkan dalam slot. Jumlah *bye* didapat dari jumlah slot dikurangi total jumlah peserta. *Placeholder* ini akan berada di urutan paling bawah jika tabel peserta diurutkan sesuai *seed*, sehingga akan menempati slot lawan tanding peserta unggulan. Setelah peserta unggulan dan *bye* menempati urutan, selanjutnya peserta non-unggulan akan diacak dan ditempatkan dalam slot.



```
// Fungsi Linear Congruential Generator (LCG)
private function shuffleWithLCG(Collection $fighters): Collection
{
    $count = $fighters->count();
    $a = 1664525; // pengali
    $c = 1013904223; // increment
    $m = pow(2, 32); // modulus
    $seed = now()->micro;

    $randomIndices = [];
    for ($i = 0; $i < $count; $i++) {
        $seed = ($a * $seed + $c) % $m;
        $randomIndices[] = ['index' => $i, 'rand' => $seed];
    }

    // urutkan berdasarkan nilai random
    usort($randomIndices, fn($a, $b) => $a['rand'] <=> $b['rand']);

    // susun ulang collection berdasarkan random order
    $shuffled = collect();
    foreach ($randomIndices as $ri) {
        $shuffled->push($fighters[$ri['index']]);
    }

    return $shuffled;
}
```

Gambar 24. Fungsi *linear congruential generator* atau *LCG*

Di dalam fungsi *insertBye()* tadi terdapat bagian pengacakan untuk peserta non-unggulan atau unseeded pada Gambar 23. Fungsi ini menggunakan metode *linear congruential generator* atau disingkat *LCG* yang merupakan *pseudorandom number generator* paling umum digunakan [15]. Persamaan matematika dari fungsi pada Gambar 24 dimuat dalam (2).

$$x_{n+1} = (a * x_n + c) \bmod m \quad (2)$$

dengan :

a = pengali atau multiplier

c = *increment*

m = modulus

Menurut (Bikos et al., 2023), periode maksimum untuk *LCG* adalah sama dengan *m*.

Namun, untuk mendapatkan batas atas ini, kondisi berikut harus dipenuhi:

- Parameter *c* harus relatif prima terhadap *m*.
- Jika *m* kelipatan 4, maka *a* - 1 juga harus kelipatan 4.
- Untuk semua pembagi prima *p* dari *m*, nilai *a* - 1 harus kelipatan *p*.



2. Uji SUS (*System Usability Scale*)

Pengujian SUS dilakukan dengan memberi 10 pertanyaan kepada pengguna. Setiap pengguna merupakan seorang pemain bulutangkis yang memahami cara kerja bagan pertandingan eliminasi tunggal. Penyebaran kuesiioner melalui media sosial WhatsApp dan distribusi langsung kepada pengguna di Gedung Badminton Tanujayan. Setiap pertanyaan dijawab menggunakan skala penilaian ‘Sangat Tidak Setuju’, ‘Tidak Setuju’, ‘Ragu-ragu’, ‘Setuju’, dan ‘Sangat Setuju’ [17]. Keseluruhan nilai didapat dari rata-rata nilai dari setiap pengguna sebagaimana dalam (3).

$$\begin{aligned}
 SUS = & ((Q1 - 1) + (5 - Q2) + (Q3 - 1) + (5 - Q4) + (Q5 - 1) \\
 & + (5 - Q6) + (Q7 - 1) + (5 - Q8) + (Q9 - 1) \\
 & + (5 - Q10) * 2.5)
 \end{aligned}
 \tag{3}$$

Tabel 3. Pertanyaan Kuesioner Uji SUS

Isi pertanyaan	
Q1	Saya berpikir akan menggunakan aplikasi ini lagi
Q2	Saya merasa aplikasi ini rumit untuk digunakan
Q3	Saya merasa aplikasi ini mudah digunakan
Q4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan aplikasi ini
Q5	Saya merasa fitur-fitur aplikasi ini berjalan dengan semestinya
Q6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada aplikasi ini)
Q7	Saya merasa orang lain akan memahami cara menggunakan aplikasi ini dengan cepat
Q8	Saya merasa aplikasi ini membingungkan
Q9	Saya merasa tidak ada hambatan dalam menggunakan aplikasi ini
Q10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan aplikasi ini

Hasil kuesioner SUS disajikan pada Tabel 4 bersama dengan total skor dan hasil rata-ratanya. Persamaan yang digunakan untuk menghitung rata-rata menggunakan (4).

$$\text{Rata-rata} = \frac{\sum x}{N}
 \tag{4}$$

Dengan :

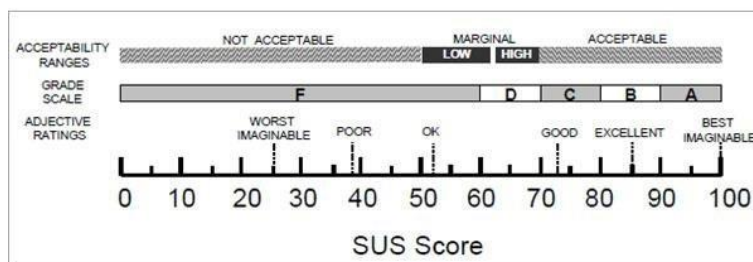
x = Total skor responden

N = Jumlah responden



Tabel 4. Hasil kuesioner SUS pada 25 responden

No	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Jumlah (x)	Skor (x*2.5)
1	4	2	4	2	4	4	3	2	2	4	23	57,5
2	4	1	5	2	4	2	5	2	4	3	32	80
3	5	1	5	1	4	2	5	1	2	1	35	87,5
4	5	1	5	2	4	2	4	2	4	2	33	82,5
5	4	1	5	1	5	3	4	1	5	3	34	85
6	5	1	5	1	4	2	5	1	5	1	38	95
7	4	1	4	1	4	1	4	1	4	3	33	82,5
8	5	2	4	1	4	1	5	1	5	1	37	92,5
9	5	1	5	1	5	1	4	1	5	4	36	90
10	4	2	5	1	4	2	3	1	4	1	33	82,5
11	4	2	4	2	4	2	4	2	4	5	27	67,5
12	5	2	5	2	5	2	5	1	5	3	35	87,5
13	2	2	4	2	4	1	3	2	5	4	27	67,5
14	5	2	4	4	4	2	4	2	3	3	27	67,5
15	4	2	5	1	4	2	4	1	5	4	32	80
16	5	2	5	2	5	2	5	2	5	2	35	87,5
17	4	3	3	4	4	3	3	3	3	5	19	47,5
18	4	2	5	2	5	1	4	1	5	2	35	87,5
19	5	1	5	1	5	1	5	1	5	5	36	90
20	4	2	4	3	4	2	4	2	4	3	28	70
21	5	3	3	3	4	3	5	3	4	4	25	62,5
22	5	2	4	3	3	1	5	1	5	4	31	77,5
23	4	2	4	2	5	2	4	1	4	4	30	75
24	5	3	4	2	4	2	4	2	4	3	29	72,5
25	3	2	4	1	4	2	4	2	4	2	30	75
Jumlah											1950	
Rata-rata											78	



Gambar 25. Skala penilaian SUS



Sebagaimana pada Tabel 4, aplikasi TurneyGen mendapat skor 78. Skor ini berada pada rentang C (70-80) yang bisa dilihat pada Gambar 25 sebagai hasil 'Good' atau baik dan 'Acceptable' atau dapat diterima. Ini menunjukkan bahwa aplikasi sudah memenuhi keinginan pengguna dan layak digunakan.

Kuesioner SUS menunjukkan bahwa 92% responden (23 orang) berminat untuk menggunakan aplikasi TurneyGen di lain waktu, dengan 1 orang netral dan 1 orang lagi tidak berminat dilihat dari hasil kolom Q1. Data responden pada Q1 berbanding lurus dengan Q3 yang juga menunjukkan bahwa 92% responden (23 orang) merasa bahwa aplikasi ini mudah digunakan, dengan 2 orang lainnya menjawab netral. Namun meskipun begitu, hasil Q10 dengan pernyataan "Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan aplikasi ini" tidak terlalu membersamai hasil dari Q1 dan Q3. Data kolom Q10 menunjukkan bahwa 40% responden (10 orang) merasa harus membiasakan diri, 7 orang menjawab netral dan 8 orang lainnya tidak perlu berada pti untuk menggunakan aplikasi TurneyGen. Dapat dikatakan bahwa aplikasi ini terasa mudah digunakan setelah pengguna beberapa kali mencoba melakukan sesuatu pada aplikasi ini. Didukung data dari kolom Q6 yang menunjukkan bahwa 84% responden (21 orang) merasa aplikasi sudah berjalan dengan konsisten dan serasi pada setiap fitur nya dengan 3 jawaban netral dan 1 orang merasa aplikasi TurneyGen tidak berjalan dengan konsisten. Pemberian tutorial yang baik seharusnya bisa membuat pengguna lebih cepat memahami bagaimana aplikasi TurneyGen dimanfaatkan dengan baik.

KESIMPULAN

Menilik dari hasil dan pembahasan, Sistem informasi dan undian pertandingan badminton sesuai aturan PBSI berhasil memenuhi kebutuhan pengguna. Web aplikasi TurneyGen berhasil membuat bagan otomatis yang memenuhi standar aturan PBSI dan BWF. Penempatan *byes* dan urutan *seed* dapat mengikuti tabel arahan dari BWF. TurneyGen juga telah melewati uji *blackbox* dengan memenuhi semua kesesuaian hasil. Selain itu web aplikasi TurneyGen juga memenuhi ekspektasi pengguna dalam hal kesesuaiannya. Dapat dilihat dari hasil uji SUS yang menunjukkan bahwa TurneyGen mendapat nilai 'Baik' dan 'Dapat diterima'.

UCAPAN TERIMA KASIH

Ucapan Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa, karena atas rahmat dan karunia-Nya, penulis dapat menyelesaikan penelitian yang berjudul "Sistem Informasi dan Undian Pertandingan Badminton Menggunakan Metode Kejuaraan PBSI" ini dengan baik.



Penulis menyadari bahwa tersusunnya tugas akhir ini tidak lepas dari bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Kedua orang tua tercinta, yang selalu memberikan doa, dukungan moral, dan semangat tanpa henti dalam setiap langkah yang penulis tempuh.
2. Bapak Aris Rakhmadi, S.T., M.Eng. selaku dosen pembimbing yang telah memberikan arahan, masukan, serta motivasi dalam penyusunan dan penyempurnaan tugas akhir ini.
3. Ibu Dr. Endah Sudarmilah, S.T., M.Eng. dan Bapak Dedi Gunawan, S.T., M.Sc., Ph.D. selaku dekan Fakultas Komunikasi dan Informatika dan kaprodi Teknik Informatik Universitas Muhammadiyah Surakarta yang telah memberikan kesempatan dan fasilitas selama proses perkuliahan hingga penyusunan tugas akhir.
4. Seluruh dosen pengajar dan staf akademik di lingkungan Fakultas Komunikasi dan Informatika Universitas Muhammadiyah Surakarta, yang telah membekali penulis dengan ilmu pengetahuan dan pengalaman berharga.
5. Teman-teman seperjuangan yang telah menjadi tempat berbagi semangat dan saling membantu selama proses pengerjaan tugas akhir.
6. Semua pihak yang tidak dapat penulis sebutkan satu per satu, namun telah memberikan bantuan baik secara langsung maupun tidak langsung dalam penyusunan tugas akhir ini.

Penulis menyadari bahwa tugas akhir ini masih jauh dari sempurna. Namun, penulis berharap karya ini dapat memberikan manfaat dan menjadi sumbangsih kecil bagi pengembangan ilmu pengetahuan.

DAFTAR PUSTAKA

- [1] M. H. Siregar and D. I. Mulyana, "Teknologi Artificial Intelligence (AI) Vision Swift dalam Optical Flow," 2024. Accessed: Oct. 08, 2024. [Online]. Available: <https://journal.stmiki.ac.id>
- [2] A. Nugroho and U. Sunan Bonang Tuban, "Match Fixing Pertandingan Sepak Bola Dalam Perspektif Hukum Pidana," *INNOVATIVE: Journal Of Social Science Research*, vol. 4, pp. 7506–7514, 2024, Accessed: Oct. 10, 2024. [Online]. Available: <https://j-innovative.org/index.php/Innovative/article/view/8754>
- [3] BWF, "GENERAL COMPETITION REGULATIONS DEFINITIONS," 2024. Accessed: Oct. 07, 2024. [Online]. Available: <https://extranet.bwf.sport/docs/document-system/81/1466/1471/Section%205.1%20->



%20General%20Competition%20Regulations%20-%2028%20April%202024%20V5.0.pdf

- [4] BWF, "General Competition Regulations Part III-Section 1A General Competition Regulations PART III SECTION 1 A GENERAL COMPETITION REGULATIONS 1. PURPOSE AND APPLICABILITY OF THE REGULATIONS 1.1 With these Regulations the BWF in accordance with its Rules: 1.1.1 controls the game, from an international aspect, in all countries; and 1.1.2 makes Regulations for all international competition." Accessed: Oct. 07, 2024. [Online]. Available: https://system.bwfbadminton.com/documents/folder_1_81/Regulations/GCR/Part%20III%20-%20Section%201A%20-%20General%20Competition%20Regulations.pdf
- [5] A. F. A. Aziz and M. Maryam, "Implementasi Metode Waterfall Pada Rancang Bangun Sistem Informasi Manajemen Aset Di SMK Muhammadiyah 3 Karanganyar," *Jurnal Ilmiah SINUS*, vol. 22, no. 1, p. 71, Jan. 2024, doi: 10.30646/sinus.v22i1.793.
- [6] I. A. Afwa and Y. S. Nugroho, "Sakti-ums: Sistem Aplikasi Karyawan Terbaik Ums Dengan Metode Simple Additive Weighting (SAW)," 2024, Accessed: Jul. 18, 2025. [Online]. Available: <https://eprints.ums.ac.id/121750/>
- [7] M. Abbas, R. Rioboo, C.-B. Ben-Yelles, and C. F. Snook, "Formal Modeling and Verification of UML Activity Diagrams (UAD) with FoCaLiZe," 2020. Accessed: Oct. 20, 2024. [Online]. Available: https://scholar.google.fr/citations?view_op=view_citation&hl=fr&user=ze9wV_QAAAJ&citation_for_view=ze9wV_QAAAAJ:Y0pCki6q_DkC
- [8] W. Agustina, N. Cahyani, and H. S. Abstrak, "RANCANG BANGUN SISTEM PENCATATAN HASIL PEMERIKSAAN DEMENSIA MENGGUNAKAN TEKNIK MMSE BERBASIS WEB," 2023. Accessed: Oct. 07, 2024. [Online]. Available: <https://eprints.ums.ac.id/114689/>
- [9] D. Ahrizal, M. Khaerul Miftah, R. Kurniawan, and T. Zaelani, "Pengujian Perangkat Lunak Sistem Informasi Peminjaman PlayStation dengan Teknik Boundary Value Analysis Menggunakan Metode Black Box Testing," vol. 5, no. 1, 2020, Accessed: Oct. 23, 2024. [Online]. Available: <http://openjournal.unpam.ac.id/index.php/informatika73>
- [10] A. Delfrian, A. Putra, A. Rakhmadi, and M. Eng, "SISTEM MONITORING TAHFIDZUL QUR'AN BERBASIS WEB," 2024. Accessed: Oct. 07, 2024. [Online]. Available: <https://eprints.ums.ac.id/124438/>
- [11] M. Hyzy *et al.*, "System Usability Scale Benchmarking for Digital Health Apps: Meta-analysis," *JMIR Mhealth Uhealth*, vol. 10, no. 8, Aug. 2022, doi: 10.2196/37290.



- [12] M. Pradipta, "APLIKASI GENERATOR TURNAMEN SEPAKBOLA," 2022. Accessed: Oct 07, 2024. [Online]. Available: <https://dspace.uui.ac.id/handle/123456789/40823>
- [13] D. Gunawan *et al.*, "Implementasi MERN Stack pada Pengembangan Sistem Penerimaan Peserta Didik Baru," *JURNAL SWABUMI*, vol. 11, no. 2, p. 2023, 2023, Accessed: Jul. 17, 2025. [Online]. Available: https://scholar.google.co.id/citations?view_op=view_citation&hl=en&user=pcGtGbUAAAAJ&sortby=pubdate&citation_for_view=pcGtGbUAAAAJ:r0BpntZqJG4C
- [14] A. B. Praja, D. Darmansah, and S. Wijayanto, "Sistem Informasi Pencatatan Surat Masuk dan Surat Keluar Berbasis Website Menggunakan Metode Waterfall," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 3, no. 3, p. 273, Mar. 2022, doi: 10.30865/json.v3i3.3914.
- [15] Rinaldi Munir, "Matematika_Diskrit_Rinaldi_Munir," 2005, Accessed: Jul. 18, 2025. [Online]. Available: <https://archive.org/details/matematika-diskrit-rinaldi-munir/page/217/mode/2up>
- [16] A. Bikos, P. E. Nastou, G. Petroudis, and Y. C. Stamatiou, "Random Number Generators: Principles and Applications," *Cryptography*, vol. 7, no. 4, Dec. 2023, doi: 10.3390/cryptography7040054.
- [17] E. Pradana and I. C. A. Priyawati, "Android Game-Based Learning Media Recognizes the Structure and Functions of Plant and Animal Parts for Elementary School," 2022. Accessed: Jul. 17, 2025. [Online]. Available: https://scholar.google.com/citations?view_op=view_citation&hl=en&user=OG8hRvEAAAAJ&sortby=pubdate&citation_for_view=OG8hRvEAAAAJ:WF5omc3nYNoC

