



IMPLEMENTASI METODE FEATURE DRIVEN DEVELOPMENT PADA PERANCANGAN WEB PORTAL LOMBA

Pramudya Tamir Indra Permana

Teknik Informatika, Universitas Pembangunan Nasional Veteran Jawa Timur, Jl. Rungkut Madya No.1, Kec. Gn. Anyar, Surabaya, Jawa Timur, Indonesia
19081010167@student.upnjatim.ac.id

Submitted : Oktober 2023, Revised Oktober 2023, Available Online Oktober 2023

ABSTRACT

This research aims to implement Feature Driven Development in web portal competition planning using the FDD method which is part of a variation of the agile method which has a direct approach which makes the development system simpler, easier to understand and easier to implement. By using the website to access the competition participants more easily. The stages of this research go through the scheme of developing an overall model, building a list of features, planning by feature, designing by feature, and building by feature. The results can show the use of cloud server services in storing information so that data usage in the application is maintained. The cloud server uses the Supabase provider with services in the form of databases, file storage and virtual personal computers. File storage is used to store assets in the form of posters and user profile photos. Virtual private computers (VPC) are used as a place to run cloud-based servers that are integrated via API. Apart from that, this research can increase understanding of using agile methods in iterative and incremental software development.

Keywords: *Implementation, Website portal, Agile, Feature Driven Development.*

ABSTRAK

Penelitian ini memiliki tujuan untuk mengimplementasikan Feature Driven Development terhadap perencanaan web portal lomba dengan menggunakan metode FDD yang bagian dari bentuk variasi dari metode agile di mana memiliki pendekatan langsung yang menjadikan sistem pengembangan lebih sederhana, mudah dipahami, dan mudah diterapkan. Dengan menggunakan website untuk mengakses lebih mudah terhadap para peserta lomba tersebut. Tahapan penelitian ini melalui skema Developan overall model, build a features List, plan by feature, design by feature dan build by feature. Hasilnya dapat menunjukkan pada penggunaan layanan cloud server dalam menyimpan informasi agar penggunaan data pada aplikasi tetap terjaga. Cloud server menggunakan provider Supabase dengan layanan berupa database, file storage, dan virtual private computer. File storage digunakan dalam menyimpan aset berupa poster dan foto profil pengguna. Virtual private computer (VPC) digunakan sebagai tempat dalam menjalankan server berbasis awan yang terintegrasi melalui API. Selain itu, penelitian ini dapat meningkatkan pemahaman dengan menggunakan metode agile dalam pengembangan perangkat lunak yang iteratif dan inkremental.

Kata kunci: Implementasi, Website portal, Agile, Feature Driven Development.





1. PENDAHULUAN

Perlombaan adalah kompetisi untuk meraih kemenangan dengan meningkatkan kemampuan dan keterampilan. Selain bersaing dengan orang lain, ini juga berperan dalam meningkatkan ilmu pengetahuan dan prestasi (Maulana & Heryadi, 2022). Menurut analisa dari Pusat Prestasi Nasional (Puspresnas) melalui buku Rencana Strategis dijelaskan bahwa terdapat kegiatan lomba dan kompetisi yang diselenggarakan setiap tahun di setiap jenjang mulai dari tingkat Kabupaten/kota, provinsi, nasional, dan internasional sebagai upaya peningkatan prestasi. Namun, pada sebagian kota di Indonesia tidak sedikit bentuk penyajian informasi perlombaan masih menggunakan cara konvensional melalui poster atau brosur (Syahputra et al., 2019). Cara tersebut dianggap kurang efektif dan dapat menghambat proses penyebaran informasi, sehingga berimbas pada aspek seperti waktu, tenaga, hingga biaya transportasi untuk menuju lokasi pendaftaran lomba (Maulana & Heryadi, 2022). Sebagai solusi dari penyajian informasi lomba yang masih konvensional, *website* dipilih sebagai salah satu media dalam penyampaian informasi ajang perlombaan karena penggunaannya yang mudah dan dapat diakses kapan saja dan dimana saja (Aljukhadar & Senecal, 2009).

Menurut penelitian (Jessy et al., 2016) menerangkan bahwa *web portal* dapat membantu menyediakan informasi kepada pengguna dengan cara memprioritaskan informasi yang relevan bagi mereka. Pada penelitian ini, *web portal* digunakan sebagai titik temu antara peserta lomba dengan penyedia lomba. Melalui *web portal*, masyarakat yang tertarik dengan ajang perlombaan dapat mencari informasi seputar lomba dan mendaftar lomba. Rancangan sistem mereka menggunakan metode *waterfall model*. Menurut penelitian mereka, metode *waterfall* dapat memudahkan proses pengembangan dan perbaikan sistem. Meskipun demikian, menurut penelitian dengan kompleksitas sistem serupa yang dilakukan oleh (Andry et al., 2020) menyebutkan bahwa pada model tradisional seperti *waterfall*, seluruh proyek dikerjakan dalam beberapa tahap dimana setiap tahap harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap lainnya.

Salah satu bentuk implementasi dari *agile model* adalah metode *feature driven development* (FDD). Metode FDD dipilih sebagai solusi karena dapat membagi kompleksitas aplikasi berdasarkan fitur yang dikembangkan serta memiliki iterasi pengerjaan dengan jangka waktu maksimal 2 minggu untuk tiap-tiap fiturnya (Rahmad et al., 2022). FDD memiliki proses yang bersifat iteratif dan inkremental yaitu *develop an overall model*, *build feature list*, *plan by feature list*, *design by feature*, dan *build by feature* (Hariono et al., 2014).

Website

Penyebaran informasi dan komunikasi memungkinkan suatu organisasi dalam menyediakan layanan yang berkualitas tinggi bagi pengguna lainnya. Website menjadi media yang ideal dengan menjadi sarana dalam menyediakan berbagai alternatif penawaran yang dapat memuaskan kebutuhan mereka (Poddar et al., 2009). *Website* terdiri dari kumpulan halaman yang berisi dokumen atau informasi berupa berkas, gambar, video dan aset digital. Informasi tersebut umumnya disimpan dan disediakan oleh *web-server* melalui akses internet.

Web Portal

Pada konteks tersebut, *web portal* berperan sebagai media pengadaan, tempat bertukar informasi, dan penggunaan pengetahuan secara efektif (Jessy et al., 2016). Dengan demikian pengguna dapat memperoleh informasi yang tepat dan cepat, serta dapat digunakan dengan sebaik-baiknya. *Web portal* memungkinkan beberapa informasi dapat terhubung untuk keperluan yang lebih terpersonalisasi (Yalagi & Dangare, 2013). Kita dapat menganggap portal





sebagai aplikasi yang mendapatkan data dari sumber yang berbeda kemudian menampilkan produk serta proses bisnis dalam satu sektor yang sama di bawah tampilan dengan nuansa yang konsisten.

Metode Agile

Metode *agile* merupakan seperangkat metode pengembangan perangkat lunak yang iteratif dan inkremental, di mana persyaratan dan pengembangan dikerjakan dengan mengevaluasi umpan balik tentang kemajuan progres mereka dalam mengembangkan perangkat lunak, sehingga memungkinkan penyampaian progres perangkat lunak yang berkualitas tinggi untuk memenuhi kebutuhan pelanggan serta mengakomodasi perubahan persyaratan (Venkatesh et al., 2020).

Feature Driven Development

Sebuah metode pengembangan sistem yang terbukti memberikan hasil yang kontinu, nyata, dan iteratif. FDD merupakan bentuk variasi dari metode *agile* di mana memiliki pendekatan langsung yang menjadikan sistem pengembangan lebih sederhana, mudah dipahami, dan mudah diterapkan. FDD memberikan manfaat bagi pemrogram dengan menyediakan infrastruktur pendukung dan dokumentasi yang diperlukan, serta memberikan manfaat bagi pemimpin tim dan pengguna dengan memberikan informasi tentang tim dan proyek secara tepat waktu. Dengan menetapkan informasi persentase kelengkapan untuk setiap progres dapat mempermudah dalam mengidentifikasi kelengkapan suatu fitur atau kelompok fitur. Informasi tersebut juga memberikan kemajuan progres yang mudah dipahami karena bersifat non-teknis (Palmer & Felsing, 2002).

Use Case Diagram

Use case merupakan metodologi yang digunakan dalam analisis sistem untuk mengidentifikasi, mengklarifikasi, dan mengatur persyaratan sistem. Use case diagram merupakan salah satu jenis diagram berorientasi objek yang menunjukkan bagaimana suatu sistem berinteraksi dengan entitas eksternal. Ada empat simbol utama dalam use case diagram, yaitu aktor, use case, asosiasi, dan batas sistem. Aktor adalah entitas eksternal yang berperan dalam interaksi dengan sistem, use case menggambarkan urutan tindakan untuk memberikan nilai terukur kepada seorang aktor, asosiasi merupakan interaksi yang dijelaskan oleh use case, dan batas sistem merupakan bentuk persegi panjang yang menunjukkan fungsionalitas dalam ruang lingkup sistem (Aleryani, 2016).

Object Diagram

Object diagram atau diagram objek adalah sebuah representasi visual dari instance atau objek-objek yang digunakan dalam suatu sistem. Selain itu, diagram objek juga dapat membantu dalam memeriksa kesesuaian antara atribut dan hubungan antar objek dengan model class diagram yang telah dirancang sebelumnya. Diagram objek digunakan dalam pemrograman berorientasi objek dan sering digunakan dalam tahap analisis dan desain pada pengembangan perangkat lunak (Li et al., 2004).

Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku dinamis dari sebuah sistem pada tahap desain. Diagram urutan merupakan salah satu jenis diagram interaksi pada UML yang memperlihatkan interaksi antar objek dalam urutan waktu. Diagram urutan ini menunjukkan urutan pesan yang dikirim antar objek dan juga memberikan gambaran visual mengenai bagaimana objek-objek tersebut berinteraksi dalam suatu skenario. Diagram urutan digunakan pada tahap analisis dan desain pada pengembangan perangkat lunak, dan sangat berguna dalam memodelkan dan memvalidasi interaksi antar objek pada sistem yang akan dibuat (Li et al., 2004).





Web Services dan API

Web merupakan sebuah sistem berbasis *client-server*, di mana *client* dan *server* memiliki peran yang berbeda serta mereka memiliki implementasi dan penggunaan yang independen baik dibangun menggunakan bahasa atau teknologi apapun. Agar dapat melakukan komunikasi satu sama lain, umumnya pada sisi *client* menggunakan API atau *application programming interface* (Masse, 2011). API menyediakan titik akses sistem atau fungsi pustaka dengan sintaks yang terdefinisi dengan baik yang dapat diakses dari program aplikasi atau kode pengguna untuk menyediakan fungsionalitas yang terdefinisi dengan baik. Seperti yang telah dijelaskan pada teori mengenai komputasi *serverless*, segala operasi di dalam komputasi tersebut mayoritas berbasis API (Añel et al., 2020). API memiliki berbagai jenis arsitektur salah satunya yaitu REST (Safitri & Putro, 2021). REST atau *Representational State Transfer* merupakan sebuah prinsip arsitektur dalam melakukan transmisi data melalui antarmuka yang terstandarisasi seperti HTTP (*hypertext transfer protocol*) dimana merupakan sebuah protokol aplikasi untuk mendistribusikan serta berkomunikasi antara pengguna dengan internet melalui browser.

React Library dan Typescript

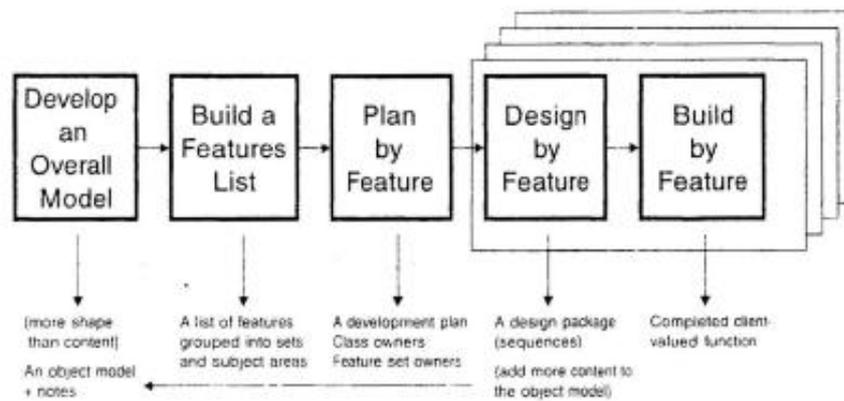
React merupakan sebuah *framework* Javascript untuk membangun antarmuka pengguna (*user interface*) pada aplikasi web yang dikembangkan oleh Facebook. React memiliki beberapa keunggulan, salah satunya yaitu komponen yang dibuat dengan React dapat digunakan kembali pada aplikasi lain dengan fungsionalitas yang sama. React juga mengkombinasikan gaya penulisan yang hampir sama dengan HTML atau biasa disebut dengan JSX, sehingga memudahkan pengembang web karena lebih familiar (Abdulloh, 2020).

Berdasarkan penjelasan yang telah dituliskan pada paragraf sebelumnya, maka fokus utama pada penelitian ini adalah melakukan perancangan *web portal* lomba menggunakan pendekatan iteratif dan inkremental dengan tujuan agar dapat menghubungkan peserta lomba dengan penyedia lomba. *web portal* lomba diharapkan juga dapat menangani logika bisnis mereka dalam menyebarkan informasi perlombaan, mengelola data pendaftar lomba hingga melakukan manajemen data lomba dalam satu wadah digital.

2. METODE

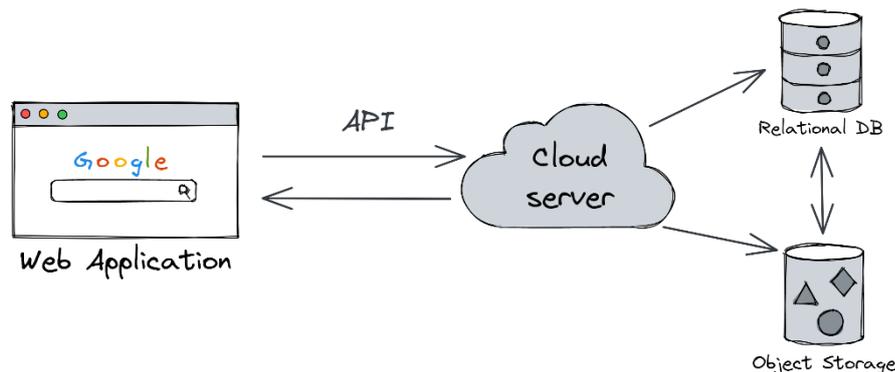
Penelitian ini menggunakan metode *Feature Driven Development* (FDD), dengan demikian penelitian ini dapat dikategorikan ke dalam tipe implementatif. Data yang digunakan pada penelitian ini sepenuhnya didapat dari studi literatur melalui perancangan sistem yang telah dilakukan oleh penelitian sebelumnya berupa data kebutuhan pengguna dan juga fitur yang akan dikembangkan dengan sedikit penyesuaian. Dengan demikian, tidak perlu adanya tahap analisis data melalui pengambilan kuesioner atau sebagainya pada awal perancangan sistem. Tahap tersebut akan tetap dilakukan pasca perancangan sistem dengan melakukan pengambilan umpan balik pengguna atas sistem yang telah dirancang dan dijalankan, sehingga pendekatan iteratif dan inkremental melalui implementasi metode FDD dapat diterapkan. Tahapan pada penelitian kualitatif ini akan menggunakan pola konsep *software development life cycle* (SDLC) model *agile* dengan pendekatan iteratif dan inkremental berupa penyampaian progres perangkat lunak yang cepat dan berkualitas tinggi untuk memenuhi kebutuhan pengguna serta mengakomodasi perubahan persyaratan. *Feature driven development* (FDD) merupakan salah satu kerangka kerja *agile* yang digunakan untuk mengatur pengembangan perangkat lunak dalam jangka waktu pendek antara satu hingga dua minggu. Tahapan penelitian akan diilustrasikan melalui skema berikut.





Gambar 1. Proses Feature Driven Development

Server yang digunakan pada penelitian ini berbasis layanan awan (*cloud service*) sehingga memungkinkan segala bentuk komputasi yang diterima atau dikirimkan akan dilakukan melalui internet. Dengan demikian, tidak ada komputer lokal yang terlibat dalam penanganan komputasi tersebut, baik basis data hingga penyimpanan lainnya disediakan oleh penyedia layanan awan. Penggunaan server berbasis layanan awan dapat memberikan berbagai keuntungan seperti skalabilitas yang lebih baik, keamanan data yang lebih terjamin, dan pengelolaan infrastruktur yang lebih mudah. Sehingga dapat menunjang pengembangan sistem melalui metode *feature driven development*.



Gambar 2. Skema Arsitektur Sistem

3. HASIL DAN PEMBAHASAN

Hasil

Penelitian ini menggunakan layanan *cloud server* dalam menyimpan informasi agar penggunaan data pada aplikasi tetap terjaga. *Cloud server* menggunakan *provider* Supabase dengan layanan berupa *database*, *file storage*, dan *virtual private computer*. *File storage* digunakan dalam menyimpan aset berupa poster dan foto profil pengguna. *Virtual private computer* (VPC) digunakan sebagai tempat dalam menjalankan *server* berbasis awan yang terintegrasi melalui API. Penelitian ini menggunakan layanan VPC Supabase berbasis *free tier* dengan spesifikasi sebagai berikut:

- *Unlimited API Request*
- *Database* sebesar 500 MB
- *Bandwidth* dengan kecepatan hingga 5GB

Prefix DOI : 10.3785/kjst.v1i6.347

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).





- File upload hingga 50MB
- Kapasitas pengguna bulanan hingga 50.000 per bulan

Pada tahap ini menjelaskan langkah dalam inialisasi sistem, pembuatan proyek aplikasi, integrasi dengan *cloud server*, konsumsi data pada *cloud server*, proses transformasi baris kode menuju bahasa mesin (*building dan compiling*), hingga mengakses aplikasi melalui *browser*.

```
→ mycode npm create vite@latest mulailomba --template react-ts
Need to install the following packages:
  create-vite@4.4.1
Ok to proceed? (y) y
├ Select a framework: > React
└ Select a variant: > TypeScript + SWC

Scaffolding project in /home/pramudya/mycode/mulailomba...

Done. Now run:

cd mulailomba
npm install
npm run dev
```

Gambar 3. Inialisasi Aplikasi Web Portal Lomba

```
npm install @supabase/supabase-js
```

Gambar 4. Instalasi Supabase Cloud Server

```
lib/supabase.ts

import { createClient } from "@supabase/supabase-js"

const SUPABASE_PROJECT_URL = "PROJECT_URL_FROM_SUPABASE"
const SUPABASE_PUBLIC_ANON_KEY = "ANON_KEY_FROM_SUPABASE"

const supabaseClient = createClient(
  SUPABASE_PROJECT_URL,
  SUPABASE_PUBLIC_ANON_KEY
)

export default supabaseClient
```

Gambar 5. Integrasi Aplikasi dengan Cloud Server

```
service/getAllLomba.ts

import supabaseClient from "lib/supabase"

function getAllLomba() {
  const allLomba = supabaseClient.from("lomba").select("*")

  return allLomba
}
```

Gambar 6. Kode Pengambilan Data Perlombaan

```
* mulailomba git:(previous) npm run build
> mulailomba@0.0.0 build
> tsc && vite build

vite v4.4.4 building for production...
transforming (1) index.htmlModule level directives cause errors when bundled, "use client" in "node_modules/react-loading-skeleton/dist/index.js" was ignored.
transforming (33) node_modules/@tanstack/react-query/build/lib/index.esm.jsModule level directives cause errors when bundled, "use client" in "node_modules/@tanstack/react-query-devtools/build/lib/index.esm.js" was ignored.
transforming (53) node_modules/@tanstack/react-query-devtools/build/lib/index.esm.jsModule level directives cause errors when bundled, "use client" in "node_modules/@tanstack/react-query-devtools/build/lib/devtools.esm.js" was ignored.
transforming (184) node_modules/@tanstack/react-query-devtools/build/lib/index.esm.jsModule level directives cause errors when bundled, "use client" in "node_modules/@tanstack/react-query-devtools/build/lib/index.esm.js" was ignored.
transforming (123) node_modules/@tanstack/react-query-devtools/build/lib/index.esm.jsModule level directives cause errors when bundled, "use client" in "node_modules/@tanstack/react-query-devtools/build/lib/Explorer.esm.js" was ignored.
v 2099 modules transformed.
asset /assets/organizer-example-5-430c8cc5.svg 4.08 KiB
asset /assets/organizer-example-2-1b7c1f2b.svg 4.08 KiB
asset /assets/organizer-example-4-48333314.svg 4.08 KiB
asset /assets/organizer-example-3-09602109.svg 3.77 KiB
asset /assets/organizer-example-0-66c76d11.svg 3.75 KiB
asset /assets/organizer-example-1-b9a95556.svg 3.28 KiB
asset /assets/top-data-e9446121.svg 3.14 KiB
asset /assets/medium-data-05846483.svg 2.87 KiB
asset /assets/inter-regular-41a8f707.ttf 204.83 KiB
asset /assets/inter-medium-18044331.ttf 214.71 KiB
asset /assets/inter-regular-494c2204.ttf 212.76 KiB
asset /assets/inter-bold-790c138b.ttf 214.10 KiB
asset /assets/inter-regular-62c73d09.ttf 414.72 KiB
asset /assets/medium-data-05846483.svg 4.06 KiB gzip: 0.48 KiB
asset /assets/index-3f34286f.css 92.64 KiB gzip: 7.85 KiB
```

Gambar 7. Transformasi Baris Kode

Prefix DOI : 10.3785/kjst.v1i6.347

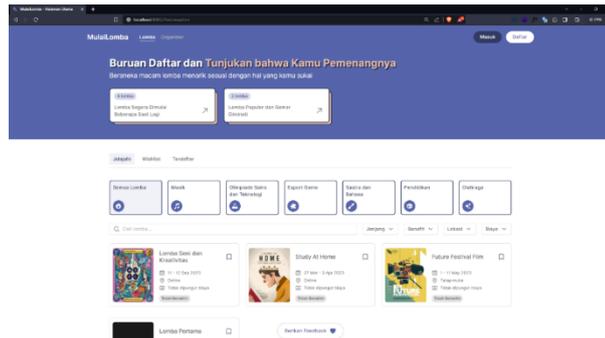
This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).





```
→ mulaiLomba git:(preview) npm run preview
> mulaiLomba@0.0.0 preview
> vite preview
→ Local: http://localhost:8080/
→ Network: use --host to expose
```

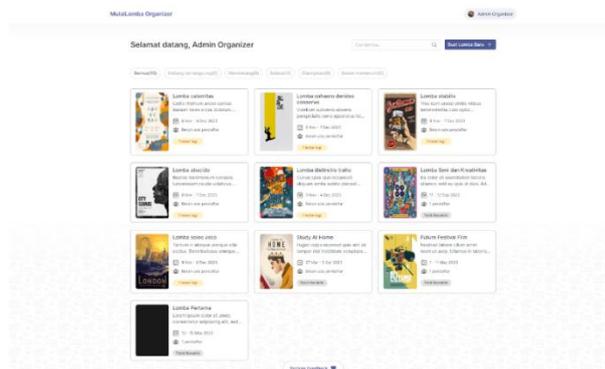
Gambar 8. Proses Menjalankan Aplikasi



Gambar 9. Cuplikan Aplikasi Web Portal Lomba

Pembahasan

Pada bagian ini akan menerangkan bentuk implementasi dari pembaharuan tampilan dasbor organizer yang telah dirancang pada bab sebelumnya. Pembaharuan tampilan yang dilakukan diantaranya berupa menambahkan aksesoris warna border, menambahkan poster pada masing-masing lomba yang telah dibuat agar mempermudah visualisasi organizer dalam mengelola lomba, dan juga menambahkan *pattern* pada *background* untuk mengurangi kesan membosankan. Implementasi pembaharuan ditunjukkan melalui Gambar 10.



Gambar 10. Pembaharuan Tampilan Dasbor Organizer

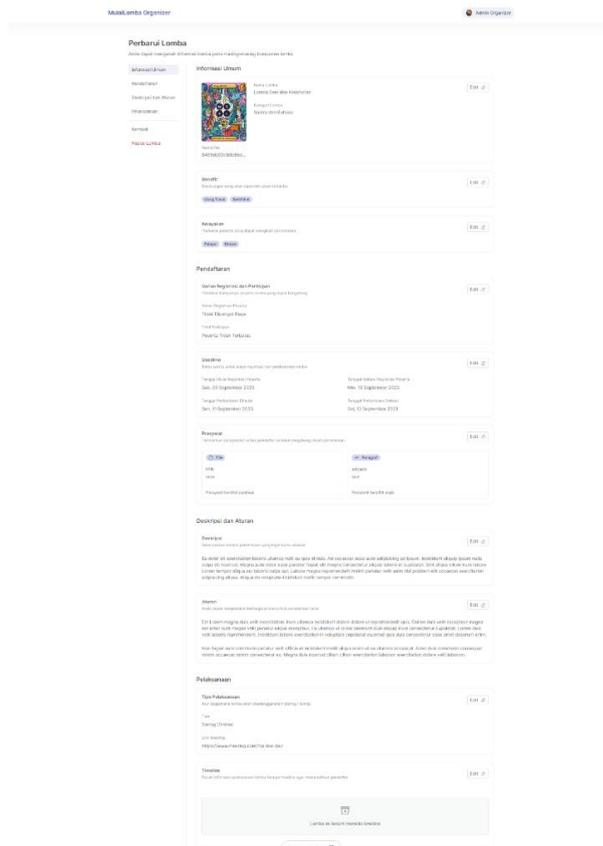
Web Portal Lomba

Pada bagian ini akan mengilustrasikan dan menjelaskan antarmuka dari aplikasi utama web portal lomba, dimana berisi proses bisnis utama dari penelitian ini. Fokus dari proses bisnis aplikasi utama terdiri dari dua buah entitas, yaitu peserta lomba dan penyedia lomba (organizer). Aplikasi memungkinkan peserta lomba untuk memperoleh informasi perlombaan dan informasi organizer, melakukan registrasi perlombaan, monitoring informasi pendaftaran lomba, hingga menyimpan lomba yang disukai.

Prefix DOI : 10.3785/kjst.v1i6.347

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).

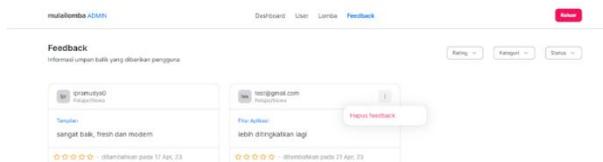




Gambar 11. Halaman Pembaharuan Lomba

Web Portal Lomba Admin

Pada bagian ini akan mengilustrasikan dan menjelaskan antarmuka dari aplikasi web portal lomba pada sisi admin. Serangkaian fitur pada aplikasi ini dibuat dengan tujuan untuk mengelola berbagai informasi yang berada di aplikasi utama pada hierarki akses yang lebih tinggi. Pada aktivitas pengelolaan data pengguna, admin dapat melakukan perubahan *email*, *password*, dan hapus pengguna. Aktivitas ini tentunya dapat dilakukan hanya ketika pengguna telah melakukan proses pendaftaran pada aplikasi utama dan tidak berlaku pada pengguna yang *login* menggunakan *Google provider*. Admin juga dapat menghapus lomba apabila dirasa merugikan suatu pihak atau tidak sesuai dengan etika hukum dan norma.



Gambar 12. Halaman Monitoring Umpan Balik

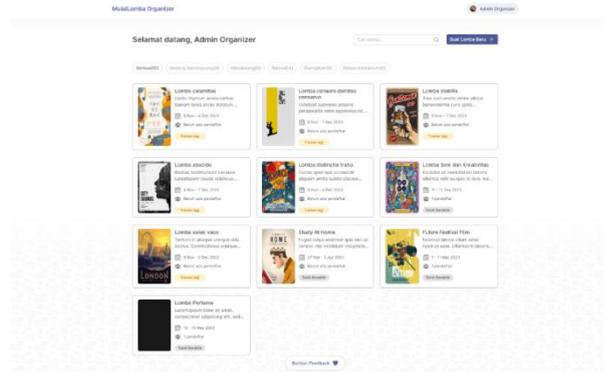
Implementasi Pembaharuan Tampilan Dasbor Organizer

Pada bagian ini akan menerangkan bentuk implementasi dari pembaharuan tampilan dasbor organizer yang telah dirancang pada bab sebelumnya. Pembaharuan tampilan yang dilakukan diantaranya berupa menambahkan aksesoris warna border, menambahkan poster pada masing-masing lomba yang telah dibuat agar mempermudah visualisasi organizer dalam mengelola lomba, dan juga menambahkan *pattern* pada *background* untuk mengurangi kesan membosankan.

Prefix DOI : 10.3785/kjst.v1i6.347

This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/).





Gambar 13. Pembaharuan Tampilan Dasbor Organizer

4. SIMPULAN DAN SARAN

Simpulan

Berdasarkan pada penelitian diatas menunjukkan bahwa dapat disampaikan oleh penulis melalui penelitian ini adalah bahwa implementasi feature driven development pada perencanaan web portal lomba dapat memberikan hasil yang sangat positif bagi para pengguna. Dalam penelitian ini, website telah berhasil diimplementasikan dengan berbagai macam tahapan penelitian ini melalui skema berikut ini: Developan overall model, build a features List, plan by feature, design by feature dan build by feature. Metode FDD dipilih sebagai solusi karena dapat membagi kompleksitas aplikasi berdasarkan fitur yang dikembangkan. Keberhasilan dalam implementasi ini dapat menjadi referensi untuk pengembangan maupun organisasi yang dapat mengadopsi berbagai macam pengembangan yang sama.

Saran

Adapun beberapa saran yang dapat diberikan untuk berbagai kebutuhan dalam penelitian tersebut yakni sebagai berikut:

1. Melakukan analisis yang baik terhadap proses pengembangan
2. Perbaiki kekurangan pada sistem website untuk menghindari error.
3. Siapkan beberapa macam rencana dalam setiap pemeliharaan dan mekanisme dukungan untuk para pengguna.

Dengan menerapkan berbagai macam saran yang dijelaskan, pelaksanaan dalam pengembangan website dengan feature driven development dapat memberikan dampak yang sangat bagus, oleh karena itu perlu diperhatikan setiap melakukan perkembangan.

DAFTAR PUSTAKA

- Abdulloh, R. (2020). *Menguasai React JS Untuk Pemula*. Self Publishing.
- Aleryani, A. Y. (2016). Comparative Study between Data Flow Diagram and Use Case Diagram. *International Journal of Scientific and Research Publications*, 6 (3), 124–127.
- Aljukhadar, M., & Senecal, S. (2009). How the Website Usability Elements Impact Performance. *M.L. Nelson et Al. (Eds.): Value Creation in e-Business Management, LNBIP 36*, 113–130. https://doi.org/10.1007/978-3-642-03132-8_10
- Andry, J. F., Tannady, H., & Gunawan, F. E. (2020). Purchase Order Information System using Feature Driven Development Methodology. *International Journal of Advanced Trends in Computer Science and Engineering*, 9 (2), 1107–1112. <https://doi.org/10.30534/ijatcse/2020/32922020>



- Añel, J. A., Montes, D. P., & Iglesias, J. R. (2020). *Cloud and Serverless Computing for Scientists*. Springer Cham. <https://doi.org/10.1007/978-3-030-41784-0>
- Hariono, M. F., Albert, Vinsensius, A., & Kusnadi, A. (2014). Developing Review Websites Using Feature Driven Development (FDD). *Ultimatics : Jurnal Teknik Informatika*, 6(2), 100–104. <https://doi.org/10.31937/ti.v6i2.338>
- Jessy, A., Rao, M., & Bhat, K. S. (2016). Web Portal: An E-Content Knowledge Management Tool. *Pearl A Journal of Library and Information Science*, 110(4), 224–233. <https://doi.org/10.5958/0975-6922.2016.00031.0>
- Li, X., Liu, Z., & Jifeng, H. (2004). A Formal Semantics of UML Sequence Diagram. *Australian Software Engineering Conference. Proceedings*, 168–177. <https://doi.org/10.1109/ASWEC.2004.1290469>
- Masse, M. (2011). *REST API Design Rulebook*. O'Reilly Media, Inc.
- Maulana, A. F., & Heryadi, R. (2022). Perancangan Sistem Informasi Perlombaan Berbasis Website untuk Kemudahan Penyampaian Informasi dan Pendaftaran Lomba. *OKTAL : Jurnal Ilmu Komputer Dan Sains*, 1(3), 263–270.
- Palmer, S. R., & Felsing, J. M. (2002). *A Practical Guide to Feature Driven Development*. Prentice Hall.
- Poddar, A., Donthu, N., & Wei, Y. (2009). Web site customer orientations, Web site quality, and purchase intentions: The role of Web site personality. *Journal of Business Research*, 62(4), 441–450. <https://doi.org/10.1016/j.jbusres.2008.01.036>
- Rahmad, S., Sofi, K., Shadiq, J., & Arifin, R. W. (2022). Selection of Feature Driven Development (FDD) Model in Agile Method for Developing Information System of Mosque Management. *Journal of Computer Networks, Architecture and High Performance Computing*, 4(2), 127–136. <https://doi.org/10.47709/cnahpc.v4i2.1469>
- Safitri, R. K., & Putro, H. P. (2021). Implementasi REST API untuk Komunikasi Antara ReactJS dan NodeJS (Studi Kasus: Modul Manajemen User Solusi247). *AUTOMATA*, 2(1).
- Syahputra, S. M., Akbar, R. A. Al, Hutabarat, F., & Setiawan, A. B. (2019). Sistem Informasi Lomba Kota Bekasi. *Software and E Bisnis*, 1(5). <https://doi.org/10.54650/jusibi.v1i5.96>
- Venkatesh, V., Thong, J. Y. L., Chan, F. K. Y., Hoehle, H., & Spohrer, K. (2020). How Agile Software Development Methods Reduce Work Exhaustion: Insights on Role Perceptions and Organizational Skills. *Information Systems Journal*, 30(4), 733–761. <https://doi.org/10.1111/isj.12282>
- Yalagi, P. S., & Dangare, C. S. (2013). Design of An Academic Web Portal Providing E-Facilities. *International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR)*, 3(1), 85–90.

