

ANALISIS MODEL MENTAL MAHASISWA DALAM PENGEMBANGAN E-LEARNING PEMROGRAMAN BERBASIS GAMIFIKASI MENGGUNAKAN METODE AGILE UX

Retno Indah Rokhmawati^{*1}, Fitra A. Bachtiar², Fajar Pradana³, Kharis Alfian⁴

^{1,2,3,4}Universitas Brawijaya, Malang

Email: ¹retnoindahr@ub.ac.id, ²fitra.bachtiar@ub.ac.id, ³fajar.p@ub.ac.id, ⁴kharisalfian@student.ub.ac.id

^{*}Penulis Korespondensi

(Naskah masuk: 06 Oktober 2021, diterima untuk diterbitkan: 21 Februari 2022)

Abstrak

Tujuan penelitian ini adalah menghasilkan model mental mahasiswa tentang bagaimana perilaku mereka belajar materi pemrograman. Model mental ini kemudian diterapkan dalam perbaikan desain *e-learning* CodeManiac. CodeManiac sebelumnya telah dikembangkan di Fakultas Ilmu Komputer Universitas Brawijaya (FILKOM UB) untuk mendukung matakuliah pemrograman di tingkat perguruan tinggi. Penelitian ini diawali dengan mengujicobakan CodeManiac versi awal kepada 15 mahasiswa FILKOM UB yang telah diseleksi untuk merepresentasikan karakteristik lima program studi S1 di FILKOM UB. Pertimbangan yang digunakan dalam memilih 15 mahasiswa tersebut didasarkan pada penelitian sebelumnya yang membahas model mental mahasiswa dibagi menjadi 3 persona (berdasarkan model mental penelitian sebelumnya) dalam bentuk level, antara lain: level dasar (*elementary*), level sedang (*intermediate*), dan level lanjutan (*advanced*). Penelitian ini lebih menajamkan pada susunan menu/fitur yang diperlukan agar kebutuhan belajar ketiga level tersebut terakomodasi dalam satu platform, yaitu CodeManiac. Oleh karena penelitian ini telah didasari penelitian eksperimen sebelumnya dan berfokus pada perbaikan desain solusi, maka Agile UX sesuai diterapkan pada penelitian ini. Hasil penelitian ini adalah menerapkan model mental ke dalam desain baru CodeManiac, yang selanjutnya divalidasi kembali kepada 15 mahasiswa yang telah terlibat pada pengujian awal. Penelitian ini menemukan kebutuhan baru terkait *Sign Up* dan *Login*, *Course*, *Achievements*, *Exercises*, *Challenge*, dan permasalahan umum (navigasi dan *layout*). Berdasarkan pengujian desain perbaikan, 15 mahasiswa memberikan umpan balik yang menunjukkan peningkatan kualitas sebagai berikut: peningkatan sebesar 24,1% pada aspek *learnability*, peningkatan sebesar 20,5% pada aspek *intuitive efficiency*, peningkatan sebesar 42% pada aspek *ease of use*, dan peningkatan sebesar 56% pada aspek *interactivity*.

Kata kunci: Agile UX, CodeManiac, E-Learning, Model Mental, Pemrograman Dasar.

STUDENTS' MENTAL MODEL ANALYSIS IN DEVELOPMENT OF E-LEARNING GAMIFICATION-BASED PROGRAMMING USING AGILE UX METHOD

Abstract

The purpose of this research is to analyze students' mental models of how their behavior learns programming material. This mental model is then applied to improve the *e-learning* CodeManiac design. CodeManiac has previously been developed at the Faculty of Computer Science Universitas Brawijaya (FILKOM UB) to support programming courses at the university level. This research begins by testing the initial version of CodeManiac to 15 students of FILKOM UB who have been selected to represent the characteristics of five undergraduate study programs at FILKOM UB. The considerations used in selecting the 15 students were based on previous research that discussed the mental model of students divided into 3 personas (based on previous research) in the form of levels, including elementary level, intermediate level, and advanced level. This research is more focused on the arrangement of the menus/features needed so that the learning needs of the three levels are accommodated in one platform, namely CodeManiac. Because this research has been based on previous experimental research and focuses on improving solution design, Agile UX is appropriate to be applied to this research. The result of this research is to apply the mental model into the new CodeManiac design, which is then validated again to 15 students who have been involved in the initial test. This research finds new needs related to Sign Up and Log in, Courses, Achievements, Exercises, Challenges, and common problems (navigation and layout). Based on the improvement design test, 15 students gave feedback indicating the quality improvement as follows: 24.1%

improvement in learnability aspect, 20.5% improvement in intuitive efficiency aspect, 42% improvement in ease-of-use aspect, and 56% improvement on the aspect of the interactivity.

Keywords: *Agile UX, CodeManiac, E-Learning, Mental Model, Programming*

1. PENDAHULUAN

Pemrograman menjadi salah satu materi yang dianggap sulit oleh mahasiswa sejak tahun pertama (Yatim et al., 2012). Mahasiswa tingkat pertama seharusnya dapat memahami dan mendeskripsikan konsep pemrograman. Tetapi sebagian besar mahasiswa hanya mampu menyelesaikan pertanyaan dasar dan menghafal sintaks tanpa memahami artinya (L. Shackelford, 2006). Du Boulay mencatat lima alasan terkait kesulitan mahasiswa dalam belajar pemrograman: (1) untuk apa pemrograman dan keuntungan apa yang didapat dalam pembelajaran pemrograman; (2) bahasa pemrograman yang dianggap sebagai model abstrak dalam bidang komputasi; (3) sintaksis dan semantik bahasa pemrograman; (4) struktur program; dan (5) pragmatik (keterampilan dalam perencanaan, analisis, pengembangan, pengujian, *debugging*) (Bower dan Falkner, 2015) (Prabawa et al., 2018). Kelima alasan inilah yang menjadi inspirasi penelitian ini untuk merancang *e-learning* yang dikembangkan berdasarkan aktivitas alami dan perilaku target pengguna lalu menggambarannya dalam model mental. Model mental dalam bidang UI/UX (*User Interface/User Experience*) didefinisikan sebagai sebuah model untuk merepresentasikan sudut pandang pengguna terkait sistem yang digunakan. Dari model mental tersebut dapat menggambarkan apa saja yang dibayangkan, diketahui, dan diyakini oleh pengguna terkait sistem yang digunakan. Hal ini dapat memudahkan pengembang sistem dalam menemukan *insight* (aspek penting) dan ciri khas (terutama untuk kepentingan *branding*) (Payne, 2008). Dengan menerapkan model mental, peneliti dapat berempati secara mendalam, sehingga memberikan gambaran perilaku yang stabil dari segmen pengguna yang akan diteliti. Melalui model mental, peneliti dapat mengetahui motivasi yang paling mendasar dari sudut pandang pengguna (Young, 2008).

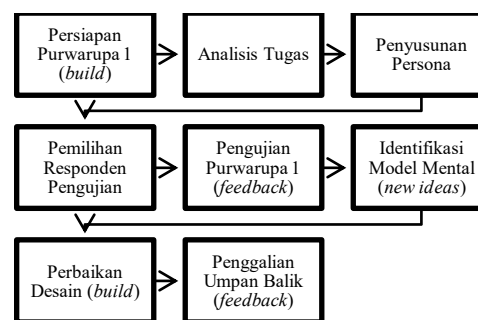
Penerapan model mental dapat digunakan untuk meningkatkan strategi pembelajaran sehingga dapat meningkatkan keberhasilan belajar (Bower dan Falkner, 2015) (Rokhmawati dan Az-Zahra, 2019). Penelitian ini merupakan pengembangan lanjut dari penelitian oleh Rokhmawati dan Az-zahra (2019) dengan melibatkan 200 mahasiswa FILKOM UB sebagai responden menunjukkan data bahwa 67% mahasiswa menyadari bahwa materi pemrograman sangat penting dan 27% mahasiswa berpendapat materi pemrograman penting. Secara motivasi dan keminatan, mahasiswa memiliki respon yang positif terhadap materi pemrograman. Hasil survei tersebut juga menunjukkan bahwa 52% mahasiswa menemui

kesulitan memahami materi sejak tahun pertama, yaitu materi Pemrograman Dasar. Berdasarkan survei tersebut, muncul kebutuhan mahasiswa selaku target pengguna membutuhkan *e-learning* yang membantu mahasiswa belajar mandiri dan mendukung mahasiswa dalam mengeksplorasi kemampuan sesuai dengan kecepatan belajar mahasiswa (*learner-paced*).

FILKOM UB telah mengembangkan *e-learning* berbasis gamifikasi yang disebut CodeManiac (dapat diakses melalui <https://codemaniac-filkom.ub.ac.id/>). Penelitian ini bertujuan untuk menguji purwarupa dan menggali model mental mahasiswa dengan menggunakan CodeManiac versi awal ini. Output penelitian ini berupa hasil analisis model mental yang diterapkan pada perbaikan desain antarmuka CodeManiac. Guna mendapatkan saran perbaikan yang dibutuhkan hingga menghasilkan purwarupa yang valid dan adaptif diterapkanlah metode Agile UX. Agile UX sesuai diterapkan pada penelitian yang telah didasari oleh produk dari eksperimen sebelumnya. Sehingga pelaksanaan proses Agile UX sudah didasari gambaran target pengguna dan kebutuhan yang lebih spesifik.

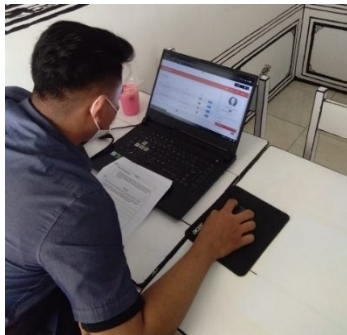
2. METODE PENELITIAN

Tahapan Agile UX pada penelitian ini diawali dengan purwarupa yang sudah tersedia kemudian dieksplor untuk mendapatkan umpan balik dari target pengguna, kemudian dianalisis menjadi ide baru untuk memperbaiki desain solusi. Gambar 4 menjelaskan tentang metode penelitian yang akan dilakukan dalam pelaksanaan penelitian ini. Penelitian ini diawali dengan mempersiapkan purwarupa awal yang dapat diakses melalui <https://codemaniac-filkom.ub.ac.id/>. Metodologi pada Gambar 1 diadaptasi dari proses Agile UX yang terdiri dari tahapan pengembangan (*build*), pengumpulan umpan balik pengguna (*feedback*), dan penemuan ide baru (*new ideas*).



Gambar 1. Metode Penelitian

Dari purwarupa CodeManiac tersebut ditentukan skenario tugas/*task* yang berfokus pada alur dasar belajar pemrograman dasar (Tabel 1). Selanjutnya, tim peneliti menyeleksi responden dari mahasiswa untuk mewakili lima program studi S1 yang ada di FILKOM UB dan dilibatkan dalam pengujian purwarupa secara luring. Namun, tim peneliti hanya mendapatkan 15 responden yang berkomitmen mewakili tiga program studi. Pemilihan responden tersebut didasarkan pada persona dengan menggunakan model mental, antara lain: level dasar (*elementary*), level sedang (*intermediate*), dan level lanjutan (*advanced*). Penempatan level ini didasari nilai hasil pretest para responden mahasiswa. Mahasiswa dipilih dengan mempertimbangkan pengalaman belajar materi pemrograman dasar dari ketiga level yang sudah dijelaskan pada landasan teori.



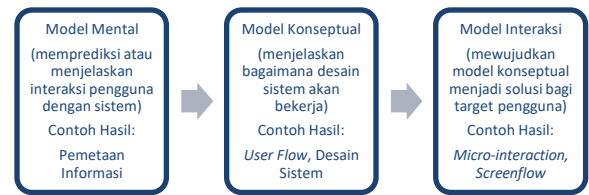
Gambar 2. Kondisi Pengujian Purwarupa CodeManiac Awal

Pengujian purwarupa awal dilakukan secara luring dibagi menjadi 2 sesi dengan tetap menerapkan protokol kesehatan COVID-19 (Gambar 2). Pengujian ini bertujuan untuk mengidentifikasi masalah dan model mental baru yang berfokus pada arsitektur informasi. Tujuan dalam tahap ini adalah mendapatkan pemahaman tentang masalah dan memunculkan empati dari peneliti terhadap permasalahan tersebut (Dam dan Siang, 2019). Solusi terbaik muncul akibat pemahaman yang baik terhadap perilaku pengguna. Lebih jauh, perlu juga digali emosi yang mendorong perilaku pengguna. Sehingga dapat muncul solusi yang inovatif dapat memenuhi kebutuhan pengguna, baik kebutuhan yang disadari atau tidak (Doorley et al., 2018). Tahap selanjutnya adalah melakukan perbaikan desain CodeManiac dan mengujikannya kembali pada 15 responden yang sama dengan pengujian awal.

3. LANDASAN TEORI

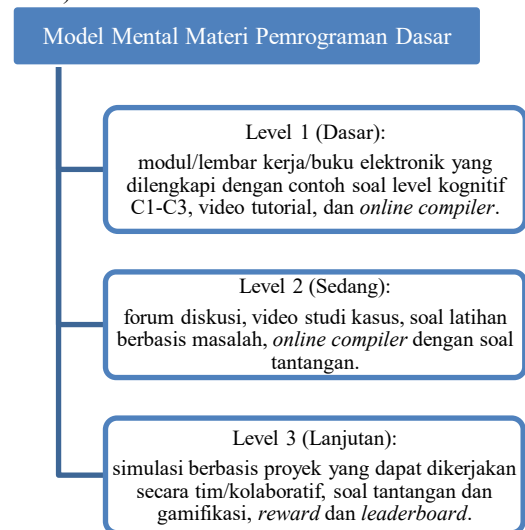
3.1 Model Mental Materi Pemrograman

Model mental merupakan salah satu tahapan dalam UX *research* yang menjadi dasar dalam perancangan model konseptual yang kemudian menjadi model interaksi (Gambar 3). Model mental dirancang berdasarkan data kualitatif dari segmen wawancara, observasi, *card sorting*, dan teknik sejenis lainnya.



Gambar 3. Alur Proses Model Mental Menjadi Model Interaksi
Diadaptasi dari: (Jenny Preece, Helen Sharp, 2019)

Model mental seseorang sangat dinamis dan berubah seiring dengan perubahan pengalaman dan konteks (Suhandi et al., 2018). Level 1, model dianggap sebagai "mainan" atau "salinan" realitas. Model yang dibuat pada level ini bukan berdasarkan pada konsep dan hanya berdasarkan pada pengalaman yang tidak dapat menjelaskan masalah (Suhandi et al., 2018).



Gambar 4. Model Mental Materi Pemrograman
Sumber: (Rokhmawati dan Az-Zahra, 2019)

Level 2, model dapat mengarah pada tujuan secara eksplisit dan spesifik (Jansoon, R.K dan E, 2009). Pada level ini, peserta didik mulai mengerti konteks permasalahan yang lebih kompleks, sehingga peserta didik lebih tertantang untuk menyelesaikan *bug* atau *error*. Level 3, model telah dibangun untuk mengembangkan dan menguji suatu ide dan dapat dimanipulasi dan dioperasikan terhadap tes. Peserta didik pada level ini, mampu menyelesaikan persoalan konseptual, kontekstual, dan bahkan *problem-solving*. Ketiga level ini menjadi penting karena sebagai dasar pengembangan fitur pembelajaran yang disediakan pada CodeManiac versi awal untuk mengakomodasi variasi kemampuan mahasiswa dalam satu *platform*.

3.2 Agile UX

Agile UX dalam penelitian ini berfungsi sebagai metode pengembangan purwarupa yang menerapkan model mental dari penelitian sebelumnya. Agile UX dipilih karena sesuai dengan kondisi penelitian yang telah didahului dengan penelitian eksperimen untuk

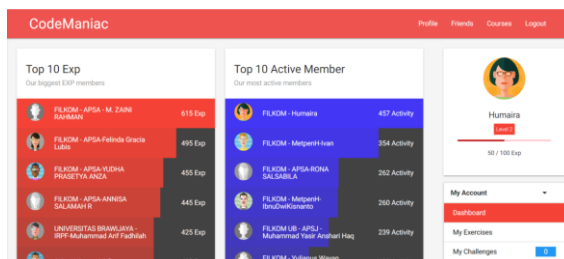
mendapatkan model mental materi pemrograman dasar (Gambar 5). Agile UX umumnya didefinisikan sebagai proses analitik dan kreatif yang melibatkan seseorang untuk bereksperimen, membuat purwarupa, mengumpulkan umpan balik, dan mendesain ulang suatu antarmuka sebuah sistem (Zaharias dan Mehlenbacher, 2012). Pada penelitian sebelumnya menyatakan bahwa Agile UX dapat menghubungkan kebutuhan desain suatu organisasi dengan inovasi yang ingin dicapai oleh organisasi tersebut (Bruun et al., 2018). Pada penelitian yang dilakukan Sabrina Bresciani menyatakan bahwa Agile UX tidak hanya mengeksplor desain visual semata, tetapi juga ranah kognitif (model mental), komunikasi, dan kolaborasi (Bresciani, 2019)(González-González, Toledo-Delgado dan Muñoz-Cruz, 2015). Saat ini Agile UX sering diterapkan pada *setting* pendidikan untuk dapat menghasilkan solusi yang kreatif. Namun metode Agile UX dapat berkembang sangat luas dengan iterasi yang tidak terbatas sehingga membutuhkan waktu penelitian yang lama. Untuk mengatasi hal tersebut, peneliti harus menentukan batasan masalah yang akan dijadikan fokus (Melles, Howard, & Thompson-Whiteside, 2012)(Kieffer, Ghouti dan Macq, 2017). Keunggulan Agile UX adalah metode yang sesuai untuk menyelesaikan permasalahan desain yang kompleks (Kieffer, Ghouti dan Macq, 2017) (Schwartz, 2014) dengan iterasi yang dilakukan berdasarkan data yang berkembang saat diskusi dilakukan.



Gambar 5. Alur Agile UX.
Sumber: (Landis, 2016)

4. HASIL DAN PEMBAHASAN

Penelitian diawali dengan mempersiapkan purwarupa awal CodeManiac. Purwarupa ini dikembangkan berdasarkan kebutuhan fungsional dasar seorang mahasiswa yang belajar materi pemrograman dasar dan dilengkapi dengan fitur gamifikasi dengan nama *MyChallenge* yang menampilkan pencapaian mahasiswa pada fitur *Leaderboard* (Gambar 6).



Gambar 6. Tampilan Fitur *Leaderboard* CodeManiac Awal

Tahap analisis skenario tugas menghasilkan alur tugas yang dilalui oleh semua responden (Tabel 1). Alur disusun berdasarkan perilaku pengguna saat belajar materi pemrograman dasar.

Tabel 1. *Task* untuk pengguna

Tugas ke-	Deskripsi Tugas
1	Registrasi dan masuk dengan akun
2	Melihat kategori materi dan mengambil <i>course</i>
3	Melihat <i>achievements</i>
4	Melihat <i>exercises</i>
5	Mengedit <i>profile</i>
6	<i>Challenge</i> bersama teman

Masing-masing responden menuliskan pengalamannya saat pengujian pada sebuah lembar kerja. Penelitian ini berfokus pada respon kualitatif yang mengindikasikan permasalahan, tingkat stres, dan tingkat keberhasilan yang dirasakan oleh responden (Tabel 2).

Berdasarkan Tabel 2, dapat dikategorikan menjadi tiga kondisi. Pertama, tugas yang mudah seperti tugas 1, 3, dan 5 yang ditunjukkan dengan respon 100% responden tidak stres dan 100% berhasil menyelesaikan tugas. Kedua, tugas sederhana namun aspek pragmatismenya masih membingungkan seperti tugas 4. Ketiga, tugas yang cukup menantang namun masih bisa diselesaikan oleh responden seperti tugas 2 dan 6. Berdasarkan hasil pengujian, terdapat permasalahan mayor pada tugas 2 dan 6 terkait susunan materi dan navigasi fitur *challenge*. Hasil respon pada Tabel 2 dibuat menjadi model mental yang menunjukkan susunan fitur yang intuitif menurut responden. Permasalahan dikategorikan berdasarkan halaman/fitur dan permasalahan umum (Gambar 7).

Tabel 2. Identifikasi Permasalahan

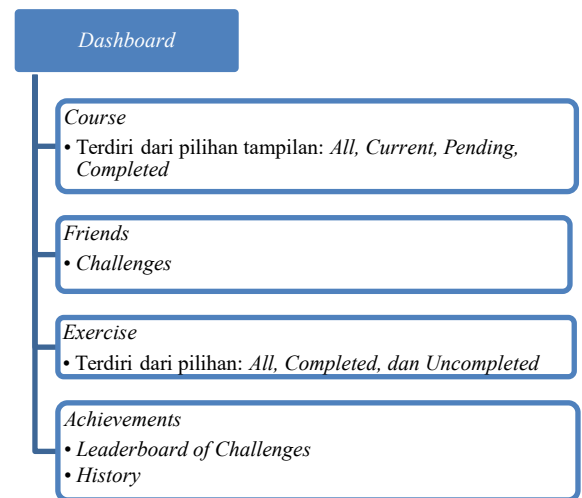
No.	Fitur, Tingkat Stress, Tingkat Keberhasilan	Permasalahan
1	Sign Up dan Login (100% Tidak Stres, 100 Berhasil)	Halaman <i>Sign Up</i> dan <i>Login</i> belum menampilkan <i>brand</i> atau logo identik dari CodeManiac.
2	<i>Course</i> (26% Tidak Stres, 74% Berhasil)	Susunan <i>Course</i> membingungkan. Belum memperhatikan hierarki visual yang baik. Perlu ada pengkategorian materi yang sudah diselesaikan dan materi yang belum diselesaikan. Belum ada keterangan jumlah sub topik dalam setiap materi.
3	<i>Achievements</i> (100% Tidak)	Tampilan <i>Achievement</i> kurang memberi kesan memuaskan.

No.	Fitur, Tingkat Stress, Tingkat Keberhasilan	Permasalahan
	Stres, 100 Berhasil)	Perlu dibuat <i>badge</i> agar lebih menarik secara visual.
4	<i>Exercises</i> (93% Tidak Stres, 93% Berhasil)	Belum ada keterangan atau filter untuk melihat latihan yang sudah atau yang belum diselesaikan, sehingga pengguna harus mencari satu-persatu.
5	<i>Challenge</i> (100% Tidak Stres, 100 Berhasil)	Belum ada filter atau tab untuk memilah challenge yang sudah atau yang belum diselesaikan.
6	Permasalahan Umum (20% Tidak Stres, 100% Berhasil)	<ul style="list-style-type: none"> <i>Sidebar</i> navigasi sebaiknya di sebelah kiri untuk memudahkan penggunaan. Susunan navigasi perlu ditata hierarkinya. Karena navigasi CodeManiac saat ini kurang intuitif berada di sebelah kanan, sehingga kurang diperhatikan keberadaannya.

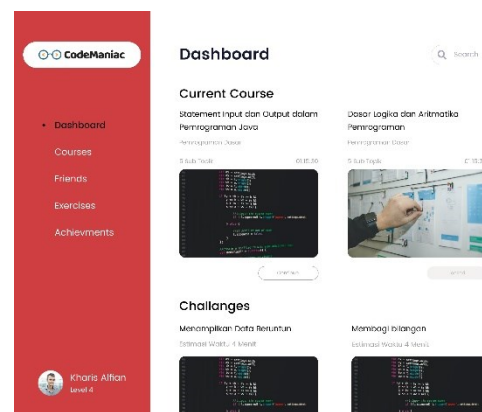
Berdasarkan Gambar 7, terdapat perubahan penamaan navigasi/fitur yang semula terdiri dari: *Course*, *Achievement*, *Exercise*, *Profile*, dan *Challenge*, kemudian diubah menjadi *Dashboard* (menampilkan gambaran umum e-learning, materi terkini, dan soal tantangan terbaru), *Course* (menampilkan materi), *Friends* (berisi soal tantangan/challenge), *Exercise* (berisi soal latihan), dan *Achievement* (menampilkan pencapaian dan level pengguna pada materi pemrograman). Perubahan signifikan yang berbeda antara Gambar 6 dan Gambar 8, CodeManiac baru telah memiliki logo *branding*, sidebar terdapat di sebelah kiri, dan perubahan navigasi sesuai dengan model mental Gambar 7.

Pada halaman *dashboard* dibagi menjadi *Sidebar* (navigasi sebelah kiri) dan *Content* (dua kolom *Course*). Dari hasil wawancara, warna *branding* CodeManiac perlu dipertahankan, yaitu warna merah. Hal ini mendukung kontras antar komponen, sehingga pengguna mudah fokus untuk melihat dan memahami cara penggunaan. Pada Gambar 8 sudah ditambahkan logo CodeManiac yang berupa huruf dan dikombinasikan dengan simbol tag (< >) yang identik dengan kegiatan belajar pemrograman. Penggunaan warna merah memberikan kontras yang baik bagi pengguna kesan *emphasizing* pada saat berinteraksi dengan CodeManiac. Untuk memberikan kelegaan dalam membaca, konten *Course* dibuat 2 kolom dengan visual hierarki berupa Judul Materi, Kategori Materi, Informasi Sub Topik, Gambar *Course*, dan tombol untuk mengakses materi.

Secara umum, desain dasar sama dengan halaman *dashboard* awal. Desain halaman selanjutnya merupakan repetisi desain dengan tetap menampilkan *sidebar* merah dan logo CodeManiac. Desain halaman *Exercises* (Gambar 9) yang berbeda dari *Dashboard* awal adalah adanya pengkategorian latihan menjadi *All* (semua materi yang diakses mahasiswa), *Completed* (materi yang sudah terselesaikan), *Uncompleted* (materi yang sudah diakses namun belum diselesaikan).



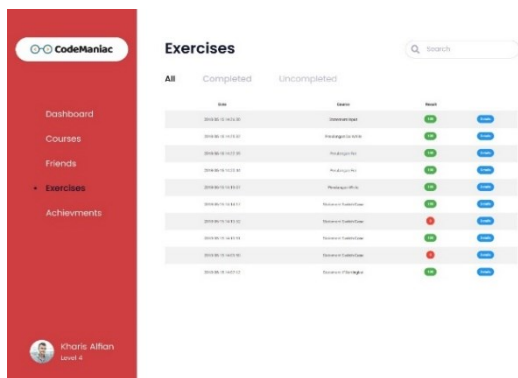
Gambar 7. Model Mental untuk Perbaikan Arsitektur Informasi CodeManiac



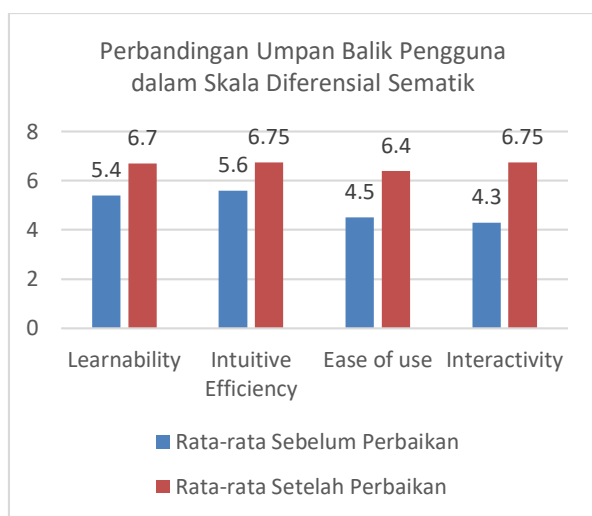
Gambar 8. Tampilan *Dashboard*

Pengkategorian ini juga diterapkan pada menu *Challenge* dan *Course* untuk memudahkan pengguna dalam memilah konten/materi yang belum atau yang sudah diselesaikan. Hasil purwarupa yang telah diperbaiki diujikan kembali kepada 15 responden yang telah dilibatkan sebelumnya. Perbandingan umpan balik antara purwarupa lama dan baru dapat dilihat pada Gambar 10.

Pertanyaan umpan balik terdiri dari 4 poin, antara lain: *learnability*, *intuitive efficiency*, *ease of use*, dan *interactivity* yang diadaptasi dari teori *pedagogical usability* (Nokelainen, 2006)(Jahnke et al., 2018). Umpan balik diekspresikan dalam skala diferensial semantik dengan rentang 1-7. Berdasarkan Gambar 10, pengujian kedua menunjukkan peningkatan sebesar 24,1% pada aspek *learnability*, peningkatan sebesar 20,5% pada aspek *intuitive efficiency*, peningkatan sebesar 42% pada aspek *ease of use*, dan peningkatan sebesar 56% pada aspek *interactivity*.



Gambar 9. Tampilan Exercise



Gambar 10. Perbandingan Umpan Balik Pengguna

5. KESIMPULAN

Model mental dapat membantu proses pengembangan produk pendukung pembelajaran dalam menemukan fokus perbaikan berdasarkan perspektif target pengguna. Model mental dapat digunakan sebagai dasar penentuan persona agar mendapat gambaran kriteria responden yang merepresentasikan segmen pengguna. Model mental juga dapat digunakan sebagai gambaran peta situs, sehingga susunan materi/arsitektur informasi menjadi lebih intuitif bagi pengguna/peserta didik. Penerapan model Agile UX sesuai diterapkan pada proses pengembangan produk pendukung pembelajaran asalkan sudah didukung penelitian/eksperimen sebelumnya untuk menemukan fokus konteks permasalahan, dalam hal ini model mental level target pengguna. Untuk memvalidasi hasil perbaikan, dapat dilakukan dengan membandingkan hasil kuesioner *pedagogical usability* antara pengujian awal dan akhir. *Pedagogical usability* merupakan kriteria pengembangan usability dasar yang dikembangkan sesuai dengan konteks pendidikan yang erat dengan desain instruksional. Masalah penelitian ini dapat diselesaikan dengan penerapan model mental penelitian sebelumnya untuk menentukan kriteria pengguna dan menemukan model mental baru untuk menyusun arsitektur informasi *e-learning*. Hal ini

divalidasi dan dibuktikan dengan peningkatan pada keempat aspek *pedagogical usability*.

6. ACKNOWLEDGMENT

Penelitian ini didukung sepenuhnya oleh Lembaga Penelitian dan Pengabdian Masyarakat Universitas Brawijaya melalui pendanaan Hibah Peneliti Pemula tahun 2020. Peneliti juga ingin berterima kasih kepada Fakultas Ilmu Komputer yang telah memberikan dukungan sarana dan prasarana penelitian. Selain itu, ucapan terima kasih ditujukan bagi para mahasiswa program studi S1 Pendidikan Teknologi Informasi, S1 Sistem Informasi, S1 Pendidikan Teknik Informatika, dan S1 Teknologi Informasi yang telah terlibat secara aktif dalam penelitian ini.

DAFTAR PUSTAKA

- BOWER, M. DAN FALKNER, K., 2015. Computational thinking, the notional machine, pre-service teachers, and research opportunities. In: D. D'Souza dan K. Falkner, ed. *Proceedings of the 17th Australasian Computing Education Conference (ACE 2015)*, Conferences in research and practice in information technology. [daring] Australia: Australian Computer Society.hal.37–46. Tersedia pada: <<http://crpit.com/confpapers/CRPITV160Bower.pdf>>.
- BRESCIANI, S., 2019. Visual Design Thinking: A Collaborative Dimensions framework to profile visualisations. *Design Studies*, [daring] 63, hal.92–124. Tersedia pada: <<https://doi.org/10.1016/j.destud.2019.04.001>>.
- BRUUN, A., LARUSDOTTIR, M.K., NIELSEN, L., NIELSEN, P.A. DAN PERSSON, J.S., 2018. The role of UX professionals in agile development: A case study from industry. *ACM International Conference Proceeding Series*, (October), hal.352–363.
- DAM, R. DAN SIANG, T., 2019. *Test Your Prototypes: How to Gather Feedback and Maximise Learning | Interaction Design Foundation*. Interaction Design Foundation.
- DOORLEY, S., HOLCOMB, S., KLEBAHN, P., SEGOVIA, K. DAN UTLEY, J., 2018. *Design Thinking Bootleg*.
- GONZÁLEZ-GONZÁLEZ, C.S., TOLEDO-DELGADO, P. DAN MUÑOZ-CRUZ, V., 2015. Metodologías ágiles centradas en personas para desarrollar software educativo. *DYNA (Colombia)*, 82(193), hal.187–194.
- GROSSLIGHT, L., UNGER, C., JAY, E. DAN SMITH, C.L., 1991. Understanding models and their use in science: Conceptions of middle and high school students and experts. *Journal of Research in Science Teaching*, [daring] 28(9),

- hal.799–822. Tersedia pada: <<http://doi.wiley.com/10.1002/tea.3660280907>>.
- JAHNKE, I., SCHMIDT, M., PHAM, M. DAN SINGH, K., 2018. Sociotechnical-Pedagogical Usability for Designing and Evaluating Learner Experience in Technology-Enhanced Environments. (c).
- JANSOON, N., R.K, C. DAN E, S., 2009. Understanding Mental Models of Dilution in Thai Students. *International Journal of Environmental & Science Education. International Journal of Environmental and Science Education*, 4(2), hal.147–168.
- JENNY PREECE, HELEN SHARP, Y.R., 2019. *Interaction Design: beyond human-computer interaction (5th edition)*.
- KIEFFER, S., GHOUTI, A. DAN MACQ, B., 2017. The Agile UX Development Lifecycle: Combining Formative Usability and Agile Methods. *Proceedings of the 50th Hawaii International Conference on System Sciences (2017)*, hal.577–586.
- L. SHACKELFORD, R. et al, 2006. Computing Curricula 2005: The Overview Report. *ACM SIGCSE Bulletin 2006*, hal.456–457.
- LANDIS, D., 2016. *What does Lean UX have that I don't? [Part 1 of 3] | Lithespeed*. [daring] <https://lithespeed.com/>. Tersedia pada: <<https://lithespeed.com/lean-ux-dont-part-1-3-2/>>.
- NOKELAINEN, P., 2006. An empirical assessment of pedagogical usability criteria for digital learning material with elementary school students. *Educational Technology and Society*, 9(2), hal.178–197.
- PAYNE, S.J., 2008. *Mental Models in Human-Computer Interaction. The Human-Computer Interaction Handbook*.
- PRABAWA, H.W., SUTARNO, H., KUSNENDAR, J. DAN RAHMAH, F., 2018. Learning basic programming using CLIS through gamification. *Journal of Physics: Conference Series*, 1013(1).
- ROKHMAWATI, R.I. DAN AZ-ZAHRA, H.M., 2019. Identifying students' mental model for Java programming subject. *ACM International Conference Proceeding Series*, (8), hal.165–169.
- SCHWARTZ, L., 2014. Agile-User Experience Design: Does the Involvement of Usability Experts Improve the Software Quality? State of the Art and a First Experiment. *International Journal on Advanced in Software*, 7(3), hal.456–468.
- SUHANDI, A., SAMSUDIN, A., RUSDIANA, D., WIBOWO, F.C., SUPRIYATMAN, S. DAN MANSYUR, J., 2018. Design of Experimental Problem Solving-Based Learning Program to Improve Mental Model and to Enhance Mental-Modeling Ability. *Jurnal Pendidikan Fisika Indonesia*, 14(2), hal.73–82.
- YATIM, N.F.M., RAHMAN, R.A., LATIH, R., ZAINAL, N.F.A., RAHMAT, M. DAN SHAHRANI, S., 2012. Students' Perception and Motivation Towards Programming. *Procedia - Social and Behavioral Sciences*, 59, hal.277–286.
- YOUNG, I., 2008. *Mental Models: Aligning design strategy with human behavior. Ubiquity*, .
- ZAHARIAS, P. DAN MEHLENBACHER, B., 2012. Exploring User Experience (UX) in virtual learning environments. *International Journal of Human Computer Studies*, 70(7), hal.475–477.

Halaman ini sengaja dikosongkan