

Penerapan Behavior Driven Development Untuk Uji...

By: SITI NADIAH HIJRIYANI

As of: Mar 8, 2022 2:27:44 PM
4,940 words - 23 matches - 21 sources

Similarity Index

6%

Mode: Similarity Report ▼

paper text:

Jurnal Tugas Akhir Fakultas Informatika Januari 2022 Penerapan Behavior Driven Development Untuk Uji Validasi (Studi Kasus Proyek Radya Labs: Sistem Informasi Binus School) Siti Nadiah Hijriyani¹, Sri Widowati², Dana Sulistyio Kusumo³

[1,2,3Fakultas Informatika, Universitas Telkom, Bandung](#) 1sitinadahh@

3

[students.telkomuniversity.ac.id, 2sriwidowati@telkomuniversity.ac.id](#) , 3danaksumo@

[telkomuniversity.ac.id Abstrak](#)

Sebelum perangkat lunak diserahkan ke pengguna, diperlukan pengujian untuk memastikan fitur yang dibuat sesuai dengan kebutuhan, maka dilakukan pengujian validasi. Dalam penelitian ini, proses uji validasi akan dilakukan menggunakan metode Behavior Driven Development (BDD) dengan mendefinisikan requirement ke dalam skenario step by step sesuai perilaku pengguna terhadap perangkat lunak. Objek yang diuji adalah proyek sistem informasi untuk Binus School di Radya Labs. Pada proses pengujian ini juga akan dibuat automated testing yang terintegrasi BDD menggunakan tools Katalon Studio. Hasil dari penelitian ini, menunjukkan bahwa pengimplementasian BDD dapat dilakukan pada sistem informasi sekolah. Dampak yang paling terlihat adalah pendeskripsian requirement yang jauh lebih terarah dan terperinci. Kata kunci : Behavior Driven Development (BDD), uji validasi, katalon studio Abstract Before the software is handed over to the user, testing is needed to ensure the features created are in accordance with the needs, then validation testing is carried out. In this study, the validation test process will be carried out using the Behavior Driven Development (BDD) method by defining requirements into step by step scenarios according to user behavior towards the software. The object being tested is an information system project for the Binus School at Radya Labs. In this testing process, automated testing that is integrated with BDD will also be made using the Katalon Studio tools. The results of this study indicate that the implementation of BDD can be carried out in school information systems. The most visible impact is the description of requirements that is much more focused and detailed. Keywords: Behavior Driven Development (BDD), uji validasi, katalon studio 1. Pendahuluan Latar Belakang Pada pengembangan perangkat lunak, sebelum perangkat lunak diserahkan ke pengguna diperlukan

pengujian untuk memastikan fitur **yang** dibuat **sudah sesuai dengan kebutuhan**

7

pengguna serta memastikan perangkat lunak dapat berjalan sesuai dengan fungsionalitasnya, maka dari itu perlu dilakukan uji validasi (validation test). Uji validasi merupakan tahap terakhir dalam pengembangan perangkat lunak sebelum sampai ke pengguna. Tujuan uji validasi untuk

memberikan jaminan akhir bahwa aplikasi yang dirancang telah memenuhi kriteria yang

4

dibutuhkan [3]. Pengujian validasi pada sebuah perangkat lunak biasanya dilakukan secara manual dengan mengacu pada test case yang didefinisikan berdasarkan poin-poin requirement yang sudah ditentukan diawal oleh tim pengembang dan klien [15]. Pengujian ini masih memiliki kelemahan pada bagian pendefinisian requirement yang terkadang tidak mencakup semua kebutuhan, dikarenakan requirement yang sudah ditentukan berbentuk kalimat yang menggambarkan hasil yang ingin dicapai. Bentuk ini tidak menggambarkan secara menyeluruh tentang proses dan kondisi yang diinginkan, sehingga sering terjadi ketidaksesuaian pemahaman requirement antara tim pengembang dan klien (contohnya ketika pembuatan fitur master level yang tidak boleh ada duplikasi data). Maka, dibutuhkan sebuah pendekatan untuk membuat requirement perangkat lunak dapat didefinisikan secara jelas, terperinci dan dapat dimengerti oleh semua stakeholder. Pendekatan tersebut yaitu Behavior Driven Development (BDD). BDD merupakan sebuah pendekatan pengembangan perangkat lunak yang mempunyai fokus pada pendefinisian requirement dengan cara membuat berbagai skenario step-by-step dari perilaku user. Pembuatan skenario akan dilakukan secara terperinci

dan menggunakan bahasa yang mudah dipahami oleh semua

21

stakeholder [14]. Sehingga sangat cocok untuk dipakai dalam pengujian validasi karena karakteristiknya yang membuat requirement dalam bentuk skenario penggunaan aplikasi. Sehingga penguji dapat lebih mudah membuat test case yang sesuai dengan requirement. Pada penelitian ini, pengujian validasi dilakukan pada proyek yang sedang dikerjakan di Radya Labs yaitu Sistem Informasi Binus School. Pengujian ini menerapkan Behavior Driven Development dengan mengkombinasikan automated testing menggunakan tools Katalon Studio. Katalon studio digunakan untuk mengeksekusi test case berdasarkan file behavior driven development yang telah dibuat. Topik dan Batasannya Fokus pada penelitian ini adalah penerapan Behavior Driven Development pada tahap uji validasi di proyek pembuatan sistem informasi Binus School. Selain itu akan dilakukan juga pembuatan automated testing yang mengacu pada skenario BDD dengan menggunakan Katalon Studio. Dengan begitu

rumusan masalah pada penelitian ini adalah sebagai berikut: a. Bagaimana melakukan

5

identifikasi skenario untuk pengujian validasi perangkat lunak dengan menggunakan pendekatan Behavior Driven Development? b. Bagaimana cara mengimplementasikan Behavior Driven Development ke dalam automated testing? Batasan penelitian ini hanya mencakup implementasi BDD pada tahap uji validasi dan fitur yang ada pada modul subject document. Tujuan Tujuan pada penelitian ini yaitu menerapkan Behavior Driven Development (BDD) untuk uji validasi pada aplikasi Binus School. Diharapkan proses pengujian dapat

sesuai dengan kebutuhan yang telah ditentukan di awal

19

Behavior Driven Development (BDD) Behavior Driven Development (BDD) merupakan metode agile

16

yang mendukung kolaborasi antara pemangku kepentingan bisnis, developer dan semua pihak yang terkait dalam proyek. Fokus utama BDD adalah menuliskan skenario yang menjelaskan behavior (kebiasaan) sistem dari perspektif pengguna dan mendorong kolaborasi antar stakeholder [5, 13]. Keuntungan BDD adalah metode ini menyediakan bahasa umum berdasarkan kalimat sederhana dan terstruktur yang memfasilitasi komunikasi antara stakeholder. Hal ini tentu mengurangi kesalahpahaman dan menghilangkan kode yang tidak perlu dan pemborosan fungsional [13]. Dengan menerapkan BDD berarti semua fitur memiliki skenario pengujian yang dapat memastikan alur kerja perangkat lunak sesuai dengan kebutuhan pengguna [13]. Untuk menemukan titik awal komunikasi antara klien dengan tim proyek dalam pengumpulan requirement, diperlukan identifikasi kebutuhan dan fitur yang akan dikerjakan terlebih dahulu. Sebuah fitur akan direalisasikan menjadi user story. User story menggambarkan interaksi antara pengguna dan sistem. Template yang digunakan untuk membuat user story dapat dilihat pada gambar 1. Gambar 1 Template user story Meskipun user story memberikan kemudahan dalam memahami fitur, user story masih memiliki kendala dalam penerapan. Sering terjadi ambiguitas yang disebabkan

penggunaan user story dan ruang lingkup yang dijelaskan

18

tidak terlalu jelas. Untuk mengatasi ini, user story dapat dirinci menggunakan serangkaian skenario BDD yang menggambarkan acceptance criteria [5]. Skenario menjelaskan bagaimana sistem harus berperilaku ketika dalam keadaan tertentu. Skenario ini menggunakan bahasa Domain Specific Language (DSL) atau yang dikenal dengan Gherkin [9]. Gherkin language adalah bahasa khusus yang membantu perilaku bisnis tanpa perlu menjelaskan implementasi secara detail. Di dalam Gherkin, ada tiga keyword penting yaitu Given (Initial state sebelum skenario berjalan), When (kondisi yang dilakukan pengguna untuk menjalankan sistem), Then (hasil respon dari sistem). Istilah penting di Gherkin, yaitu: a. Feature: mendeskripsikan fitur aplikasi, setiap file hanya boleh memiliki satu fitur b. Skenario: contoh yang menggambarkan aturan bisnis c. Step: mendefinisikan proses menggunakan keyword Given, When, Then, And, But d. Background: digunakan untuk memahami beberapa langkah yang diberikan e. Other keyword: mendefinisikan dengan penambahan keyword bisa berupa tags (@) atau comments (#) Gambar 2 Ilustrasi Behavior Driven Development 2.2 Pengujian Validasi Pengujian validasi

dilakukan untuk menjamin kesesuaian perangkat lunak dengan desain yang ditetapkan sebelumnya

12

. Indikator keberhasilan pengujian validasi adalah jika fungsi yang ada pada aplikasi sesuai dengan yang diharapkan pengguna [7]. Pengujian validasi merupakan pengujian terakhir sebelum aplikasi digunakan oleh pengguna. 2.3

Automated Testing Automated testing digunakan untuk menjalankan pengujian

dengan membandingkan hasil aktual dan hasil yang diharapkan

17

. Keuntungan utama dari automated testing yaitu efficiency (waktu untuk mengeksekusi test script lebih cepat dibanding pengujian manual karena dijalankan oleh tools), repeatability (pengujian test script yang sama dapat dieksekusi dengan cara yang sebelumnya tanpa mengulanginya lagi), scope of testing (menjadi sangat mudah untuk mengetahui jumlah code yang telah dicakup oleh test case), dan decrease of expenses (menggunakan automated mengurangi pekerjaan yang dilakukan manual testing) [8]. Automated testing dapat mencakup permasalahan yang tidak dapat dilakukan oleh manual testing. Karena manual testing memiliki permasalahan diantaranya terlalu banyak mengkonsumsi waktu, tidak bisa digunakan kembali, usaha yang dikeluarkan lebih besar dan terkadang beberapa kesalahan yang tidak terlihat [9]. 2.4 Katalon Studio Katalon

studio merupakan aplikasi open source yang digunakan untuk

9

automated testing. Aplikasi ini dikembangkan oleh Katalon LLC. Dapat digunakan untuk pengujian desktop, website, mobile applications dan API. Katalon dapat menggunakan bahasa groovy, java, javascript dan lain-lain. Katalon ini dapat dieksekusi di berbagai browser (chrome, mozilla firefox), IOS dan android [10]. Salah satu kelebihan Katalon yaitu mendukung pembuatan BDD yang terintegrasi dengan cucumber. Pengguna semakin mudah untuk membuat BDD karena sudah tersedia template pembuatan BDD ketika membuat feature file. Sehingga, akan mengurangi waktu dan usaha pengguna dalam membuat format BDD yang benar [11]. 3. Metodologi Penelitian Pengerjaan penelitian ini akan mengacu pada metode Behavior Driven Development (BDD). Kontribusi penerapan BDD terletak pada tahap pembuatan skenario BDD, penyusunan test case dan eksekusi. Tahap

penelitian yang dilakukan dapat dilihat pada Gambar 3. Gambar 3 Tahap Penelitian

11

3.1 Penentuan Fungsionalitas Yang Diuji Penelitian ini dilakukan pada proyek yang sedang dikerjakan Radya Labs yaitu Sistem Informasi Binus School. Seperti sistem informasi pada umumnya, sistem informasi ini adalah

sistem informasi manajemen sekolah berbasis website dan mobile app sebagai solusi administrasi terpadu untuk sekolah yang memiliki beberapa modul terintegrasi yang dapat diakses oleh semua anggota sekolah (guru, wali kelas, pegawai sekolah dan siswa) dan orang tua/wali siswa

1

[12]. Pada penelitian ini, objek yang dipakai adalah modul subject document. Modul ini memiliki banyak fitur yang dapat diakses oleh aktor tertentu dan setiap fitur yang ada pada modul ini memiliki hubungan dengan fitur yang lain. Ada 9

functional requirement yang diuji pada penelitian ini. Penjelasan dari setiap fungsionalitas dan aktor

yang terlibat dapat dilihat pada tabel 1. Tabel 1

15

Functional Requirement No. Fitur Penjelasan User yang terlibat 1. Teacher can filter list document User yang sudah login sebagai guru dapat memfilter daftar dokumen berdasarkan kriteria tertentu Teacher 2. Teacher can see list document User yang sudah login sebagai guru dapat melihat daftar dokumen yang telah dibuat 3. Teacher can fill out documents with draft status User yang sudah login sebagai guru dapat mengisi dokumen yang berstatus draft sehingga dokumen dapat direview oleh koordinator mata pelajaran 4. Teacher can edit documents with need approval (SH) status User yang sudah login sebagai guru dapat mengedit dokumen selama status dokumen tersebut masih need approval (SH) 5. Teacher can see the history of approval document User yang sudah login sebagai guru dapat melihat riwayat persetujuan dokumen 6. Subject Head can see a list of documents with need approval (SH) status User yang sudah login sebagai Subject head koordinator mata pelajaran dapat melihat daftar dokumen yang berstatus need approval (SH) 7. Subject Head can approve documents with need approval (SH) status User yang sudah login sebagai koordinator mata pelajaran dapat menyetujui dokumen yang berstatus membutuhkan persetujuan 8. Subject Head can change User yang sudah login sebagai document with need approval (SH) status to need revision koordinator mata pelajaran dapat memberi revisi dokumen sehingga guru dapat mengupdate kembali isi dokumen 9. Head of department can approve documents with need approval (HOD) status User yang sudah login sebagai Head of department kepala departemen dapat menyetujui dokumen yang berstatus membutuhkan persetujuan Pada modul subject document terdapat tiga aktor yang mempunyai hak akses yang berbeda untuk setiap fitur, yaitu teacher, subject head, dan head of department. Teacher atau guru di modul ini dapat memasukkan dokumen pelajaran untuk ditinjau oleh subject head. Subject head atau koordinator mata pelajaran dapat menyetujui dan merevisi dokumen pelajaran dengan tingkatan approval level 1 yang diajukan oleh guru. Sedangkan head of department memiliki hak akses yang sama dengan subject head, ditambah dengan dapat menyetujui dan merevisi dokumen pelajaran dengan tingkatan approval level 2. 3.2 Pembuatan Skenario BDD Setelah ditentukan fitur apa saja yang diuji pada tabel 1. Selanjutnya proses penentuan skenario yang diawali dengan pembuatan user story berdasarkan perilaku pengguna dan fungsionalitas. Kemudian menentukan skenario berdasarkan setiap kemungkinan input dan output yang ada pada user story. Penjelasan dari setiap fitur beserta user story dan skenarionya

dapat dilihat pada tabel 2. Tabel 2 Skenario pengujian No . Fitur User Story Skenario 1

10

. Teacher can filter list document As a teacher, I can filter Teacher can filter by academic year the list of documents by Teacher can filter by academic year and level academic year, level, Teacher can filter by academic year, level and grade, term, subject, grade approval type so that I Teacher can filter by academic year, level, can see the list according grade and term to the filter Teacher can filter by academic year, level, grade, term and subject Teacher can filter by academic year, level, grade, term, subject and approval type Teacher can't filter by level Teacher can't filter by grade Teacher can't filter by term Teacher can't filter by subject Teacher can't filter by approval type 2 Teacher can see list document As a teacher, I can see Teacher can see list document the list of documents so I Teacher can see an empty table if the

document can fill in the document list does not exist 3 Teacher can fill out documents with draft status As a teacher, I can fill Teacher can open document page out documents with draft Teacher does not fill in one of the fields in the status so that the document document can be Teacher cancels filling out the document reviewed by the subject head and head of department Teacher can fill in all the fields in the document 4 Teacher can edit documents with need As a teacher, I can edit Teacher can edit documents documents that need Teacher cancels edit document approval (SH) approval (SH) 5 Teacher can see the history of approval document As a teacher, I can see the history of approval documents Teacher can see the history of approval documents Teacher can see detailed history 6 Subject Head can see a list of documents with need approval (SH) status As a subject head, I can see a list of documents that need approval so that the document can be approved Subject head can see list of document status need approval (SH) 7 Subject Head can approve documents with need approval (SH) status As a subject head, I can approve documents with the status of need approval (SH) Subject head can see document detail page with status need approval Subject head can cancel approve document Subject head can approve document 8 Subject Head can change document with need approval (SH) status to need revision As a subject head, I can change the status of the document to need revision so that the teacher can revise the document Subject head see pop up document need revision Subject head can cancel filling in revision notes Subject head can give revision 9 Head of department can approve documents with need approval (HOD) status As a head of department, I can approve documents that need approval status so that the document is approved Head of department can see document detail page with status of need approval (HOD) Head of department can cancel approve document Head of department can approve document

3.3 Penyusunan Test Case

Hasil dari skenario yang sudah dibuat pada tabel 2 akan menjadi panduan untuk pembuatan dokumentasi test case. Test case diperlukan

untuk memastikan bahwa setiap fitur **yang** dijalankan **sudah sesuai dengan kebutuhan**

7

. Sebuah test case yang baik mampu mencakup keseluruhan kasus uji dan memenuhi persyaratan perangkat lunak, tidak membuat test case yang berulang, dan

test case harus menghasilkan hasil yang sama setiap kali pengujian dengan **tidak** memerhatikan **siapa pengujiannya. Test case**

2

dibagi menjadi

beberapa komponen, seperti test case id, description , pre- **step** , step dan **expected result**

2

, dapat dilihat pada tabel 3. Pada umumnya format test case terdapat test data, tetapi pada pengujian ini tidak menggunakannya karena pengujian berupa behavior (perilaku) bukan data input. Penulisan pre-step, step dan expected result menggunakan bahasa gherkin dengan keyword utama yaitu GIVEN untuk menjelaskan konteks dari skenario masukan. WHEN untuk menjelaskan proses yang dilakukan oleh pengguna. THEN untuk Menjelaskan hasil proses yang sudah dilakukan oleh pengguna. Tabel 3 Test case contoh fitur teacher can filter list document Test Description Pre-Step

Step Expected Result case ID Scenario 1 Teacher can filter by academic year Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter academic year dropdown And teacher click apply button Then application displays a list based on the academic year filter Scenario 2 Teacher can filter by academic year and level Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter academic year dropdown And teacher select filter level dropdown And teacher click apply button Then application displays a list based on the academic year and level filter Scenario 3 Teacher can filter by academic Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter academic year Then application displays a list based on the academic year, level and grade filter year, level and grade dropdown And teacher select filter level dropdown And teacher select filter grade dropdown And teacher click apply button Scenario 4 Teacher can filter by academic year, level, grade and term Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter academic year dropdown And teacher select filter level dropdown And teacher select filter grade dropdown And teacher select filter term dropdown And teacher click apply button Then application displays a list based on the academic year, level, grade and term filter Scenario 5 Teacher can filter by academic year, level, grade, term and subject Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter academic year dropdown And teacher select filter level dropdown And teacher select filter grade dropdown And teacher select filter term dropdown And teacher select filter subject dropdown And teacher click apply button Then application displays a list based on the academic year, level, grade, term and subject filter Scenario 6 Teacher can filter by academic year, level, grade, term, subject and approval type Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter academic year dropdown And teacher select filter level dropdown And teacher select filter grade dropdown And teacher select filter term dropdown And teacher select filter subject dropdown And teacher select filter approval type dropdown And teacher click apply button Then application displays a list based on the academic year, level, grade, term, subject and approval type filter Scenario 7 Teacher can't filter by level Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter level Then application display field level disabled Scenario 8 Teacher can't filter by grade Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter grade Then application display field level disabled Scenario 9 Teacher can't filter by term Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter term Then application display field level disabled Scenario 10 Teacher can't filter by subject Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter subject Then application display field level disabled Scenario 11 Teacher can't filter by approval type Given teacher is logged in When teacher select document menu in sidebar And teacher select subject document menu And teacher select filter approval type Then application display field level disabled

3.4 Eksekusi dengan Automated Testing Tahap pertama automated testing adalah membuat feature file. Test case yang telah dibuat sebelumnya pada tabel 3 dipindahkan ke dalam feature file. Sebuah feature file terdiri dari satu atau lebih skenario yang menjelaskan apa yang seharusnya dilakukan oleh sistem dalam situasi tertentu. Penulisan test case pada feature

teacher can filter list document Setelah selesai membuat feature file, tahap selanjutnya adalah membuat test code dengan bahasa groovy yang berisi function-function dari tiap line di skenario BDD. Di dalam setiap function ini akan berisi code step testing yang berasal dari script hasil perekaman objek. Perekaman objek ini menggunakan teknik record web, spy web dan penambahan test object secara manual. Record web dilakukan dengan cara merekam setiap aksi berdasarkan skenario dan menghasilkan sebuah script objek. Sedangkan spy web digunakan untuk mengambil objek yang tidak terdeteksi ketika melakukan record web. Pada kasus objek yang sulit untuk diambil dikarenakan objek yang bergerak, maka dilakukan penambahan objek secara manual. Penambahan objek secara manual bisa menggunakan xpath atau dengan memasukan atribut dari tag yang akan diambil. Selanjutnya script objek akan dimasukan ke dalam setiap function. Berikut contoh test code yang sudah terintegerasi dengan BDD dilihat pada Gambar 5. Gambar 5 Script hasil perekaman pengujian Selain melakukan perekaman objek, pada beberapa skenario dibutuhkan logika perulangan, dimana ketika skenario yang membutuhkan pengecekan status dokumen di dalam tabel daftar dokumen. Misal skenario "Subject head can see document detail page with status need approval" disini subject head bisa mengakses halaman detail dokumen jika berstatus need approval. Jika test code sudah dilengkapi, maka skenario BDD dapat dijalankan. Tahap pengujian akan diulang jika terdapat perubahan pada aplikasi, baik karena adanya error dan bug, maupun karena adanya perubahan requirement dari stakeholder. Untuk test case yang sudah dibuat rekaman automate test-nya, maka pengujian dilakukan dengan menggunakan rekaman yang sudah ada. 4. Hasil dan Analisis Bagian ini berisi tentang bagaimana hasil dari automate testing yang dibuat berdasarkan skenario BDD

pada bab sebelumnya beserta analisis dari hasil pengujian

6

. 4.1 Hasil Pengujian Hasil pengujian akan dibuat otomatis menggunakan katalon. Format file dari hasil pengujian berupa file HTML yang berisi informasi tentang keberhasilan dan kegagalan dari setiap skenario di fitur yang diuji. Selain HTML, format hasil pengujian dapat disesuaikan dengan format lain sesuai kebutuhan. Hasil skenario pengujian yang berhasil akan berwarna hijau dan hasil skenario yang gagal akan berwarna merah. a. Hasil pengujian yang berhasil (Passed) Sebagian besar fitur yang diuji berstatus PASSED yang berarti berhasil. Ini menandakan sistem yang dibangun berhasil menampilkan output sesuai dengan skenario yang telah dibuat sebelumnya. Report yang dibuat berisi informasi tentang fitur beserta skenario pengujian yang dijalankan. Penjelasan untuk hasil dari setiap pengujian fitur yang berhasil dapat dilihat pada daftar di bawah. Gambar 6 Contoh report test

- Teacher can filter list document : Aplikasi berhasil menampilkan tabel yang berisi subject document sesuai dengan nilai parameter filter seperti year, level, grade, term, subject dan approval type yang telah diatur sebelumnya.
- Teacher can fill out documents with draft status : Aplikasi berhasil mengubah status document dari draft menjadi need approval pada document yang diisi dengan lengkap, dan tetap berstatus draft pada document yang diisi dengan tidak lengkap atau dibatalkan.
- Teacher can edit documents with need approval (SH) : Aplikasi berhasil menampilkan pesan sukses ketika aktor mengubah text pada form document dan menekan button submit.
- Teacher can see the history of approval document : Aplikasi berhasil menampilkan riwayat status document dari status draft sampai approved.
- Subject Head can see a list of documents with need approval (SH) status : Aplikasi berhasil menampilkan tabel yang document dengan status need approval (SH).
- Subject Head can change document with need approval (SH) status to need revision : Aplikasi berhasil menampilkan informasi tentang subject document yang dipilih, mengubah status document menjadi need revision apabila aktor

menekan button revision dan mengisi catatan perbaikan. • Head of department can approve documents with need approval (HOD) status : Aplikasi berhasil menampilkan informasi document, mengubah status document menjadi approved dan kembali ke halaman detail dokumen jika aktor menekan button cancel. b. Hasil pengujian yang gagal (Failed) Pada beberapa skenario, sistem gagal menampilkan output yang sudah ditentukan di skenario pengujian. Hal ini bisa disebabkan oleh kesalahan pengiriman data dari server, logika pada sisi front-end dan pengaruh revisi pada fitur lain. Dari sisi teknis, aplikasi gagal menampilkan suatu element HTML dengan nilai value atau text yang diharapkan, sehingga Katalon tidak bisa menemukan element yang sudah ditentukan. Skenario yang memiliki bug akan dilaporkan kepada tim terkait untuk segera dilakukan debug atau revisi. Penjelasan untuk skenario yang mengalami kegagalan dapat dilihat pada daftar di bawah. • Subject head can see document detail page with status need approval: Aplikasi gagal menampilkan informasi nama, alamat, upload, jenis kelamin, hobi dan example. Error ini disebabkan perubahan format key yang dikirim melalui REST API dari server. Error ini diperbaiki dengan cara menyamakan key pada function consume API di sisi front-end. Tampilan halaman yang mengalami error

ini dapat dilihat pada gambar 6. Gambar 7

8

contoh ke-1 skenario gagal • Teacher can see list document : Aplikasi gagal menampilkan list dokumen untuk guru. Error ini disebabkan pada sisi front-end di mana link API yang dimasukan salah, sehingga tidak ada respon dari server yang dikirim. Error ini diperbaiki dengan mengganti link api pada sisi front-end. Gambar 8 contoh ke-2 skenario gagal 4.2 Analisis Hasil Pengujian Dari sembilan fitur yang dijadikan objek pengujian, terdapat dua fitur yang mengalami kegagalan. Ini berarti sekitar 92% atau 23 skenario berhasil dijalankan tanpa mengalami error, detail dari

hasil pengujian dapat dilihat pada tabel 4 . Skenario yang

20

gagal akan langsung diperbaiki oleh pihak terkait. Tabel 4 Hasil Pengujian No. Fitur Jumlah Skenario Status Sebelum Fixing Passed Failed Status Sesudah Fixing Passed Failed 1 Teacher can filter list document 11 11 0 11 0 2 Teacher can see list document 2 1 1 2 0 3 Teacher can fill out documents with 4 4 0 4 0 draft status Teacher can edit 2 2 0 2 0 4 documents with need approval (SH) Teacher can see the 2 2 0 2 0 5 history of approval document Subject Head can 1 1 0 1 0 6 see a list of documents with need approval (SH) status Subject Head can 3 2 1 3 0 7 approve documents with need approval (SH) status Subject Head can 3 3 0 3 0 8 change document with need approval (SH) status to need revision Head of department 3 3 0 3 0 9 can documents approve with need approval (HOD) status Total 31 23 2 31 0 Pada tahap pengambilan test object, terjadi kesulitan dalam mengambil test object yang hanya muncul sementara seperti toast. Hal ini dikarenakan ketika object toast menghilang maka tag toast pada HTML pun akan hilang, sehingga Katalon tidak bisa mengambil data tag tersebut. Masalah ini dapat diatasi dengan menambahkan object test secara manual dengan memanfaatkan alat inspect element yang ada di browser untuk mendapatkan informasi tag dari atributnya. Untuk pengecekan perubahan status pada dokumen, dibuat sebuah function yang akan melakukan perulangan untuk mencari row dari tabel daftar dokumen, yang memiliki status sama dengan status awal di skenario dan menyimpan index dari row tersebut. Lalu sesudah pengisian dokumen, akan dilakukan pengecekan status pada row yang sudah tersimpan sebelumnya, apakah sama dengan status akhir pada skenario. Pembuatan function ini efektif

untuk skenario yang diharuskan mengecek perubahan element di banyak dokumen yang sama. Secara keseluruhan, implementasi behavior driven development pada automate testing di modul subject document dari proyek Binus School dapat disimpulkan berhasil. Karena semua skenario BDD yang ada bisa dijadikan automated testing dengan menggunakan tools Katalon. 5.

Kesimpulan Berdasarkan penelitian yang sudah dilakukan dapat disimpulkan bahwa: 1

13

. Behavior Driven Development (BDD) dapat diimplementasikan pada modul subject document proyek pembuatan sistem informasi sekolah dengan cara merumuskan BDD ditahap pembuatan skenario berdasarkan user story sampai dengan pengujian. 2. Penggunaan automated testing yang terintegrasi dengan BDD dapat dilakukan dengan menggunakan tools Katalon. 3. Berhasil mengimplementasikan 9 fitur dengan 25 skenario BDD. Terdapat 2 skenario failed dan 23 passed. 4. Diperlukan logika perulangan untuk mengecek perubahan status di daftar dokumen. Referensi [1] P. Ramya, V. Sindhura and P. V. Sagar, "Testing using selenium web driver," 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), Coimbatore, 2017, pp. 1-7, doi: 10.1109/ICECCT.2017.8117878. [2] I. Burnstein, Practical Software Testing: A Process-Oriented Approach. New York: Springer-Verlag, 2003, p. 134. [3] M. Nurudin, W. Jayanti, R. Dwi Saputro and M. Priadyan Saputra, "Pengujian Black Box pada Aplikasi Penjualan Berbasis Web Menggunakan Teknik Boundary Value Analysis", Jurnal Informatika Universitas Pamulang, vol. 4, p. 145, 2019. [Accessed 29 November 2020]. [4] "What is Automation Testing (Ultimate Guide to Start Test Automation)", Softwaretestinghelp.com, 2020. [Online]. Available: <https://www.softwaretestinghelp.com/automation-testing-tutorial-1/>. [Accessed: 26- Nov- 2020]. [5] C. Solis and X. Wang, "A Study of the Characteristics of Behaviour Driven Development", 2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications, 2011. Available: 10.1109/seaa.2011.76 [Accessed 25 November 2020]. [6] M. Rahman and J. Gao, "A Reusable Automated Acceptance Testing Architecture for Microservices in Behavior-Driven Development", 2015 IEEE Symposium on Service-Oriented System Engineering, 2015. Available: 10.1109/sose.2015.55 [Accessed 26 November 2020]. [7] H. Sulistyanto and A. SN, "URGENSI PENGUJIAN PADA KEMAJEMUKAN PERANGKAT LUNAK DALAM MULTI PERSPEKTIF", Komuniti: Jurnal Komunikasi dan Teknologi Informasi, vol. 6, no. 1, 2014. [Accessed 29 November 2020]. [8] E. Vila, G. Novakova and D. Todorova, "Automation Testing Framework for Web Applications with Selenium WebDriver", Proceedings of the International Conference on Advances in Image Processing, 2017. Available: 10.1145/3133264.3133300 [Accessed 30 August 2021]. [9] M. Kaur and R. Kumari, "Comparative Study of Automated Testing Tools: TestComplete and QuickTest Pro", International Journal of Computer Applications, vol. 24, no. 1, pp. 1-7, 2011. Available: 10.5120/2918-3844. [10] S. Tjandra, I. Maryati and J. Theopilus, AUTOMATED SOFTWARE TESTING FOR MULTI PLATFORMAPPLICATIONS USING KATALON STUDIO, vol. 20, no. 1, 2021. [Accessed 30 August 2021]. [11] "BDD Testing Framework (Cucumber integration)", <https://docs.katalon.com>, 2021. [Online]. Available: <https://docs.katalon.com/katalon-studio/docs/cucumber-features-file.html#add-feature-files>. [Accessed: 31- Aug- 2021]. [12] 2019. [Online]. Available: <https://simsekolah.com/>. [Accessed: 01- Aug- 2021]. [13] I. Raharjana, F. Harris and A. Justitia, "Tool for Generating Behavior-Driven Development Test- Cases", Journal of Information Systems Engineering and Business Intelligence, vol. 6, no. 1, p. 27, 2020. Available: 10.20473/jisebi.6.1.27-36. [14] U. Nugraha, S. Nurduha Robaiah and D. Rospinoedji, "TESTING THE INFORMATION SYSTEM SOFTWARE USING BEHAVIOR DRIVEN DEVELOPMENT METHOD", Journal Of Archaeology Of

sources:

- 1 36 words / 1% - Internet from 12-Nov-2021 12:00AM
simsekolah.com

- 2 24 words / 1% - Internet from 17-Aug-2021 12:00AM
medium.com

- 3 19 words / < 1% match - Internet from 27-Oct-2021 12:00AM
123dok.com

- 4 11 words / < 1% match - Internet from 02-Jan-2022 12:00AM
123dok.com

- 5 11 words / < 1% match - Internet from 20-May-2019 12:00AM
eprints.umm.ac.id

- 6 8 words / < 1% match - Internet from 10-Apr-2018 12:00AM
eprints.umm.ac.id

- 7 16 words / < 1% match - Internet from 01-Mar-2022 12:00AM
conference.stmikindonesia.ac.id

- 8 8 words / < 1% match - Internet from 02-Mar-2018 12:00AM
media.neliti.com

- 9 8 words / < 1% match - Internet from 15-Jul-2018 12:00AM
media.neliti.com

- 10 12 words / < 1% match - Internet from 07-Mar-2021 12:00AM
j-ptiik.ub.ac.id

- 11 11 words / < 1% match - Internet from 14-Aug-2021 12:00AM
docplayer.info

- 12 11 words / < 1% match - Internet from 24-Jun-2021 12:00AM
lontar.ui.ac.id

- 13 10 words / < 1% match - Internet from 28-Oct-2021 12:00AM
core.ac.uk

14 9 words / < 1% match - Internet from 10-Sep-2021 12:00AM
adoc.pub

15 9 words / < 1% match - Internet from 19-Nov-2019 12:00AM
ojs.umb-bungo.ac.id

16 9 words / < 1% match - Internet from 21-Aug-2021 12:00AM
searchsoftwarequality.techtarget.com

17 8 words / < 1% match - Internet from 11-Dec-2018 12:00AM
edoc.site

18 8 words / < 1% match - Internet from 14-Nov-2021 12:00AM
gilangrmdhn.medium.com

19 8 words / < 1% match - Internet from 19-May-2020 12:00AM
id.scribd.com

20 8 words / < 1% match - Internet from 08-Dec-2020 12:00AM
repository.uinjkt.ac.id

21 8 words / < 1% match - Internet from 30-Jun-2021 12:00AM
shellymushollia.wordpress.com
