

Penerapan *Game-Based Retrospective Starfish* pada Proses Scrum: Studi Kasus pada Pengembangan Aplikasi Zikirin

Nabil Raihan Alfarizi¹, Firman Ardiansyah¹, Auzi Asfarian¹, Ahmad Ridha¹, Dean Apriana Ramadhan¹, Endang Purnama Giri¹

¹Departemen Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Pertanian Bogor, Bogor, Indonesia

Correspondence: Nabil Raihan Alfarizi (nabilrei1@gmail.com)

Received: 14 12 23 - Revised: 19 12 23- Accepted: 20 12 23 - Published: 25 12 23

Abstrak. Tim pengembang aplikasi Zikirin menggunakan metode manajemen pengembangan perangkat lunak Scrum, tetapi masih menggunakan metode *sprint retrospective* dengan menjawab pertanyaan “Apa yang sudah berjalan dengan baik?” dan “Apa yang harus di-improve?” melalui *platform* Notion. Metode *sprint retrospective* ini tidak bisa menjelaskan aktivitas-aktivitas apa saja yang perlu dipertahankan, diperbanyak, dikurangi, dimulai, dan dihindari tim pengembang aplikasi Zikirin pada *sprint* berikutnya. Pada *sprint* keenam, penelitian ini berhasil menerapkan *game-based retrospective: Starfish* kepada tim pengembang aplikasi Zikirin secara daring melalui perangkat lunak kolaboratif Figma dan Zoom untuk melihat evaluasi *sprint* yang telah dilakukan oleh tim pengembang aplikasi Zikirin secara lebih rinci dan detail. Metode *game-based retrospective: Starfish* yang diterapkan pada *sprint* keenam ini dapat mengetahui aktivitas-aktivitas yang perlu dipertahankan, diperbanyak, dikurangi, dimulai, dan dihindari pada *sprint* selanjutnya selama proses pengembangan aplikasi Zikirin. Aktivitas yang perlu dipertahankan oleh tim adalah diskusi secara *synchronous* melalui Zoom, aktivitas yang perlu diperbanyak adalah membuat obrolan bersama anggota tim supaya lebih akrab, aktivitas yang perlu dikurangi adalah melakukan *screen sharing* permasalahan yang kurang penting, aktivitas yang perlu dimulai adalah mencari perspektif desain baru, dan aktivitas yang perlu dihindari adalah membuat *backlog* yang terlalu besar.

Kata kunci: perangkat lunak, retrospeksi, Scrum, Zikirin

Citation Format: Alfarizi, N.R., Ardiansyah, F., Asfarian, A., Ridha, A., Ramadhan, D.A., & Giri, E.P. (2023). Penerapan Game-Based Retrospective Starfish pada Proses Scrum: Studi Kasus pada Pengembangan Aplikasi Zikirin. *Prosiding Seminar Nasional Universitas Ma Chung (SENAM)*, 2023, (pp. 35-48).

PENDAHULUAN

Metodologi pengembangan perangkat lunak Agile sudah banyak digunakan dalam beberapa tahun terakhir. Salah satu contoh metode manajemen pengembangan perangkat lunak yang menggunakan metodologi Agile adalah Scrum. Scrum adalah kerangka kerja ringan yang dapat membantu suatu tim atau organisasi menyelesaikan masalah kompleks dengan solusi yang adaptif (Schwaber & Sutherland, 2020). Scrum memberikan proses yang sangat sederhana untuk mengelola proyek yang kompleks dan dapat lebih mudah dimengerti daripada proses manajemen proyek tradisional lainnya (Schwaber, 2004). Scrum tidak hanya bisa digunakan untuk tim yang bekerja secara luring, tetapi juga bisa menjadi kesuksesan untuk tim yang bekerja secara daring (Cucolaş & Russo, 2023). Terdapat lima agenda yang harus ada di dalam tim yang mengadaptasi Scrum, yaitu: (i) *sprint*; (ii) *sprint planning meeting*, (iii) *daily scrum*, (iv) *sprint review*, dan (v) *sprint retrospective* (Fowler, 2019).

Salah satu seremonial Scrum yang paling tidak dihargai di dalam tim Scrum adalah *sprint retrospective* (Przybyłek dan Kotecka, 2017) padahal *sprint retrospective* bertujuan untuk mengevaluasi proses *sprint* sebelumnya dengan cara melihat hubungan antaranggota tim dan proses pengembangan produk yang sedang dikembangkan (Ozoliņš, 2018). Di dalam *sprint retrospective*, tim Scrum membicarakan apa saja yang berjalan baik selama *sprint*, masalah yang ditemukan, dan solusi yang telah atau akan dilakukan untuk menyelesaikan masalah tersebut (Schwaber & Sutherland, 2020). Namun, melakukan *sprint retrospective* yang efektif dan menyenangkan adalah sebuah tantangan karena: (1) panduan Scrum tidak menjelaskan secara detail cara melakukan *sprint retrospective* yang baik; dan (2) jika seremonial ini dilakukan secara terus menerus dengan cara yang sama, tim menjadi tidak merasakan dampaknya dan terkesan membuang-buang waktu (Przybyłek & Kotecka, 2017).

Przybyłek dkk. (2022) menunjukkan bahwa pendekatan gim akan membantu tim yang akan melakukan *sprint retrospective* untuk mengurangi masalah yang biasanya terjadi pada saat *sprint retrospective*, seperti kurang terstrukturanya proses *sprint retrospective*, masalah yang terlalu banyak dikeluhkan oleh tim, dan antusiasme tim yang tidak seimbang. Mesquida dkk. (2016) telah mengumpulkan 83 jenis gim yang dapat digunakan untuk membantu tim dalam melakukan *sprint retrospective*. Salah satunya adalah Starfish yang juga digunakan oleh Przybyłek & Kotecka (2017) yang terbukti dapat meningkatkan efektivitas kinerja tim Scrum.

Pada *sprint* pertama sampai kelima, tim pengembang aplikasi Zikirin masih menggunakan metode *sprint retrospective* biasa dengan pertanyaan “Apa yang sudah

berjalan dengan baik?” dan “Apa yang perlu di-*improve*?”. Kedua pertanyaan ini dijawab oleh seluruh anggota tim pengembang aplikasi Zikirin melalui *platform* Notion. Tim pengembang aplikasi Zikirin menuliskan seluruh evaluasi dari setiap *sprint*-nya secara umum di kedua pertanyaan ini, tetapi tidak ada evaluasi yang spesifik terhadap poin-poin yang perlu dipertahankan, diperbanyak, dikurangi, dimulai, dan dihindari untuk *sprint* selanjutnya. Oleh karena itu, penelitian ini bertujuan untuk mengubah metode tim pengembang aplikasi Zikirin dari metode *sprint retrospective* menggunakan Notion dengan dua pertanyaan menjadi *game-based retrospective*: Starfish yang memiliki lima bagian pertanyaan yang bisa diisi oleh setiap anggota tim pengembang aplikasi Zikirin.

MASALAH

Sebagai bentuk evaluasi *sprint retrospective* tim pengembang aplikasi Zikirin yang masih menggunakan *sprint retrospective* dengan dua pertanyaan “Apa yang sudah berjalan dengan baik?” dan “Apa yang harus di-*improve*?”, maka dilakukan perubahan cara *retrospective* tim pengembang aplikasi Zikirin pada *sprint* keenam. Cara *retrospective* ini diubah dengan menggunakan *game-based retrospective*: Starfish untuk melihat lebih detail poin-poin yang harus dipertahankan, diperbanyak, dikurangi, dimulai, dan dihindari untuk *sprint* selanjutnya.

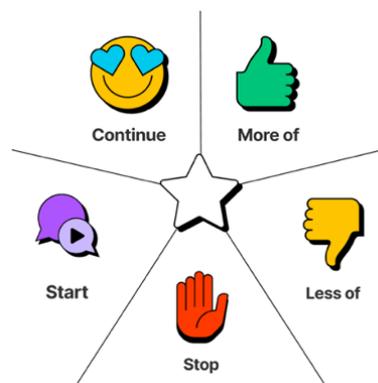
METODE PELAKSANAAN

- a. Observasi Scrum Tim Pengembang Aplikasi Zikirin Sebelum Penerapan Game-Based Retrospective: Starfish

Pada awal penelitian, langkah pertama adalah memilih tim pengembang aplikasi Zikirin sebagai subjek observasi. Tidak semua anggota tim pernah menggunakan Scrum sebelumnya. Peneliti melakukan pengamatan menyeluruh terhadap kegiatan Scrum yang dilaksanakan oleh tim sebelum metode *game-based retrospective*: Starfish. Observasi mencakup *sprint planning*, *daily standup*, pemantauan kemajuan, dan kolaborasi antaranggota tim. Aspek-aspek ini diamati untuk memahami praktik Scrum yang telah diterapkan sebelum metode baru. Observasi ini menjadi dasar pembandingan untuk menganalisis perubahan setelah penerapan metode *game-based retrospective*: Starfish. Observasi ini memberikan pemahaman mendalam tentang konteks awal tim dan perubahan yang terjadi setelah implementasi Starfish.

b. Penyusunan Strategi Game-Based Retrospective: Starfish

Setelah mengamati penggunaan Scrum oleh tim pengembang aplikasi Zikirin sebelum penerapan game-based retrospective: Starfish, langkah selanjutnya adalah merancang strategi untuk permainan tersebut. Tahap perancangan dilakukan sebelum pelaksanaan untuk menyiapkan kerangka permainan dan familiarisasi anggota tim dengan aturan permainan. Starfish retrospective dibuat dengan cara setiap anggota tim membuka fail FigJam yang sama, lalu scrum master menggandakan templat Starfish dari Staffbase sebagai kerangka awal. Gambar 1 menunjukkan templat Starfish yang dibuat oleh Staffbase. Setelah kerangka permainan selesai dan semua anggota tim bergabung di FigJam, scrum master memberikan informasi mengenai metode retrospective yang akan dilakukan, yaitu game-based retrospective: Starfish. Informasi tersebut mencakup cara bermain, alokasi waktu, dan manfaat yang dapat diperoleh melalui permainan tersebut.



Gambar 1. Templat game-based retrospective: Starfish (Staffbase & Civita, 2022)

c. Penerapan Game-Based Retrospective: Starfish pada *sprint* keenam Pengembangan Aplikasi Zikirin

Metode game-based retrospective: Starfish diimplementasikan kepada tim pengembang aplikasi Zikirin pada seremonial retrospective di *sprint* keenam. Permainan tahap pertama dilakukan dengan cara setiap individu dalam tim pengembang aplikasi Zikirin mengambil *sticky notes* dengan warna yang berbeda dan menuliskan pengalamannya pada *sprint* yang akan berakhir di setiap bagian lingkaran. Ada lima bagian lingkaran (*keep doing*, *less of*, *more of*, *stop*, dan *start*) yang bisa diceritakan oleh tiap individu di *sticky notes*-nya.

Selanjutnya, permainan tahap kedua dilakukan dengan cara setiap anggota tim pengembang aplikasi Zikirin melakukan voting untuk setiap *sticky notes* yang

telah ditulis oleh tim di lima bagian lingkaran. Masing-masing individu dapat memberikan dua suara di setiap bagian lingkaran. Setelah seluruh anggota tim melakukan voting, tim berdiskusi tentang *sticky notes* yang paling banyak dipilih oleh anggota tim dan membuat *action plan* untuk sprint selanjutnya berdasarkan *sticky notes* yang paling banyak dipilih oleh anggota tim pengembang aplikasi Zikirin.

HASIL DAN PEMBAHASAN

a. Observasi Scrum Tim Pengembang Aplikasi Zikirin Sebelum Penerapan Game-Based Retrospective: Starfish

i. *Sprint* Pertama

Dalam rapat perencanaan *sprint* pertama, tim pengembang aplikasi Zikirin menetapkan 14 *backlog* untuk diselesaikan pada akhir *sprint*. Tim bekerja bersama selama *daily scrum* dari Rabu hingga Jumat untuk menyelesaikan *backlog* yang diberikan kepada masing-masing anggota tim. Pada akhir *sprint*, seluruh *backlog* berhasil diselesaikan. Setelah itu, tim melakukan *review* dan *retrospektif sprint* pada 10 Oktober 2022. Hasil *review* mencakup beberapa kekurangan, seperti animasi *background* yang masih patah-patah dan desain Hi-Fi yang belum diberi nama komponen, menyulitkan integrasi ke dalam XCode. Retrospektif dilakukan menggunakan Notion, dengan apresiasi terhadap penyelesaian *backlog* dan optimisme tim. Beberapa perbaikan yang diusulkan meliputi memperkecil dan memperjelas *backlog*, menjelaskan *dependency task*, memperinci *acceptance criteria*, dan membuat daftar prioritas untuk setiap *backlog*.

ii. *Sprint* Kedua

Setelah menyelesaikan *sprint* pertama, tim pengembang aplikasi Zikirin memulai *sprint* kedua pada 11 Oktober 2022. Berdasarkan masukan dari retrospektif *sprint* pertama, *backlog sprint* kedua lebih banyak, mencapai 20 *backlog*. Daily Scrum tetap dilaksanakan dari Rabu hingga Jumat untuk menyelesaikan *backlog*. Seiring berjalannya *sprint*, tiga *backlog* dipindahkan ke *sprint* berikutnya karena keterbatasan waktu. Namun, tiga *backlog* lainnya tetap belum selesai karena kendala waktu dan kesulitan yang melebihi

ekspektasi, termasuk beberapa backlog yang memerlukan bantuan dari pihak luar tim.

Meskipun beberapa *backlog* belum terselesaikan, tim melanjutkan dengan sprint review dan retrospective pada 18 Oktober 2022. Hasil *review* mencakup kesulitan teknis, seperti UI Contextual Card yang tidak dapat dibuat menggunakan *design system* bawaan Apple, dan diskusi tentang penyesuaian efek suara pada audio zikir. Setelah *review*, tim melanjutkan *retrospective* menggunakan Notion. Dalam *retrospective sprint* kedua ini, ada satu anggota tim yang absen karena cuti. Tim mengakui hal-hal yang berjalan baik, seperti cepatnya penyelesaian backlog oleh Hafizh, pembuatan ERD yang sukses, dan pelaksanaan *Usability Testing* (UT) yang membutuhkan revisi. Namun, tim juga menyadari perlu perbaikan, seperti pembuatan backlog secara menyeluruh sejak awal sprint untuk memudahkan penentuan *sprint goal*, inisiasi *tech lead* untuk memperbaiki kekurangan *backlog* selama *sprint*, dan penambahan detail *backlog* bersamaan dengan penambahan tenggat waktu di tengah sprint.

iii. Sprint Ketiga

Pada 18 Oktober 2022, tim pengembang aplikasi Zikirin memulai sprint ketiga dengan sprint planning meeting untuk menetapkan sprint goal dan backlog. Terdapat 15 backlog, termasuk enam yang belum selesai dari sprint sebelumnya. Tim bekerja keras dari Rabu hingga Jumat untuk menyelesaikan *backlog*, melakukan *daily standup* setiap pagi, dan menambah satu *backlog* ke *sprint backlog*. Meskipun ditambahkan satu *backlog*, hanya delapan yang berhasil diselesaikan.

Sebelum menutup *sprint* ketiga, tim melakukan *sprint review* untuk mengevaluasi hasil pengerjaan *backlog*. Beberapa hasil *review* termasuk kurangnya kehalusan animasi *search bar* saat digulir dan perluasan gaya penulisan *copywriting* aplikasi. Setelah itu, tim melakukan *sprint retrospective* menggunakan Notion seperti *retrospective* pada *sprint* sebelumnya. Dalam *retrospective* ini, tim merasakan kemunduran kinerja karena beberapa anggota mengalami demotivasi. Beberapa *backlog* tertunda dan harus diatasi pada *sprint* berikutnya. Tim mengakui perlu peningkatan

dalam *mood management*, menjaga kesehatan, dan meningkatkan *time management* untuk meningkatkan kinerja tim pada *sprint* berikutnya.

iv. *Sprint Keempat*

Tim pengembang aplikasi Zikirin memulai *sprint* keempat pada 25 Oktober 2023 dengan delapan *sprint goals* dan 19 *backlog* hasil dari *sprint planning meeting*. *Sprint goal* ini dipengaruhi oleh masukan mentor dan hasil *sprint review* pada *sprint* sebelumnya. Dengan target peluncuran aplikasi Zikirin pada November 2023, jumlah *backlog* yang dihasilkan cukup banyak, termasuk delapan *backlog* dari *sprint* sebelumnya. Tim berhasil menyelesaikan delapan *backlog* pada Senin, 31 Oktober 2023.

Pada akhir *sprint* keempat, tim melakukan diskusi *sprint review*, membahas perbaikan untuk desain HI-FI dan kategori promotion dalam landing page aplikasi Zikirin. Setelah itu, tim melanjutkan dengan *sprint retrospective* menggunakan Notion dengan semua anggota tim berpartisipasi. Tim mengapresiasi kinerja positif saat presentasi di depan mentor dan menyelesaikan beberapa *backlog* yang menantang. Perbaikan untuk *sprint* berikutnya mencakup peningkatan referensi desain untuk desainer aplikasi Zikirin dan manajemen waktu yang lebih baik untuk pembelajaran dan pengerjaan *backlog*.

v. *Sprint Kelima*

Tim pengembang aplikasi Zikirin memulai *sprint* kelima pada 1 November 2022 dengan empat *sprint goal* dan 13 *sprint backlog* hasil *sprint planning meeting*. Sebagian besar *backlog* berasal dari *sprint* sebelumnya, dan satu *backlog* ditambahkan setelah dimulainya *sprint*, sehingga totalnya menjadi 14 *backlog*. Pada akhir *sprint* kelima, tim berhasil menyelesaikan tujuh *backlog*, sedangkan enam lainnya dipindahkan ke *sprint* berikutnya karena cakupan yang terlalu besar.

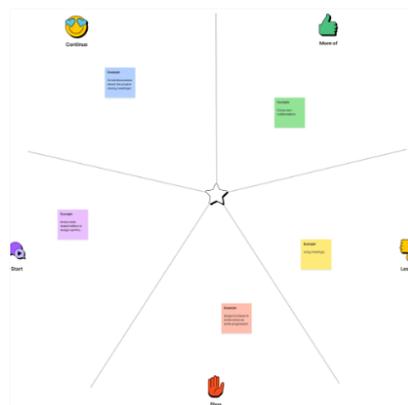
Dalam *sprint review*, tim membahas beberapa perbaikan untuk *backlog*, termasuk fitur *daily notification* yang disarankan untuk menggunakan gaya bahasa yang lebih menarik dan data zikir yang masih memiliki kekeliruan isi kontennya. Setelah itu, tim melakukan *sprint retrospective* menggunakan Notion, menggambarkan evaluasi *sprint* kelima dengan dua anggota tim yang menulis. Beberapa anggota merasa kinerja

berjalan baik, dan fokus untuk sprint berikutnya adalah meningkatkan rasa ikhlas dan intensitas komunikasi.

Selama lima *sprint*, tim selalu menggunakan metode *sprint retrospective* yang sama melalui Notion dengan dua pertanyaan umum. Metode ini tidak memberikan evaluasi yang mendalam atau rencana aksi untuk *sprint* berikutnya. Oleh karena itu, tim berencana untuk menerapkan metode *game-based retrospective: Starfish* pada akhir *sprint* keenam untuk mendapatkan evaluasi yang lebih detail dan fokus.

b. Penyusunan Strategi *Game-Based Retrospective: Starfish*

Setelah melakukan observasi Scrum sebelum penerapan *game-based retrospective: Starfish*, tahapan selanjutnya adalah penyusunan strategi untuk mempersiapkan permainan *game-based retrospective: Starfish*. Peneliti menyiapkan terlebih dahulu templat *game-based retrospective: Starfish* buatan Staffbase yang telah tersedia di galeri komunitas Figma (Staffbase & Civita, 2022). Templat ini kemudian peneliti duplikat sehingga bisa digunakan dan diedit saat *game-based retrospective* dimulai bersama dengan seluruh anggota tim pengembang aplikasi Zikirin. Templat yang sudah diduplikat dan siap dipakai dapat dilihat pada Gambar 2. Penyusunan strategi *game-based retrospective: Starfish* ini dilakukan sesaat sebelum tim pengembang aplikasi Zikirin memulai *sprint retrospective* pada *sprint* keenam.



Gambar 2. Templat *game-based retrospective: Starfish* yang sudah diduplikat

Selanjutnya, peneliti menginformasikan kepada tim pengembang aplikasi Zikirin waktu dan cara melakukan *sprint retrospective* menggunakan *game-based retrospective: Starfish*. Waktu untuk melakukan *game-based retrospective: Starfish*

sama dengan waktu yang biasanya digunakan untuk melakukan retrospective pada umumnya, yaitu sebelum sprint ditutup. Sedangkan tahapan untuk melakukan *game-based retrospective*: Starfish adalah sebagai berikut:

- i. Setiap anggota tim mengambil *sticky notes* dengan warna yang berbeda-beda antaranggota di dalam fail FigJam yang sama.
 - ii. Setiap anggota tim menulis pengalamannya masing-masing di sprint yang akan ditutup di setiap bagian lingkaran (bintang) sesuai dengan judul pada bagian tersebut.
 - iii. Setiap anggota tim memilih (dengan cara memberikan penanda) dua sticky notes di setiap bagian lingkaran (bintang) yang menurutnya harus paling diperhatikan sebagai pembelajaran di sprint selanjutnya.
- c. Penerapan Game-Based Retrospective: Starfish pada Sprint keenam Pengembangan Aplikasi Zikirin

Pada iterasi sprint keenam, tim pengembang aplikasi Zikirin mengubah metode *sprint retrospective* dengan menerapkan *game-based retrospective*: Starfish. *Sprint* ini dimulai pada 8 November 2023 dengan *sprint planning meeting* dan penyusunan *backlog* tanpa perumusan *sprint goal*. Sebanyak 11 *backlog* dibuat untuk *sprint* ini, dan setelah *daily Scrum* pada 15 November 2023, tim melanjutkan dengan *sprint review* dan *retrospective*. Dari 11 *backlog* tersebut, tiga *backlog* tidak terselesaikan karena cakupannya yang terlalu besar, menyadarkan tim akan pentingnya memecah *backlog* yang kompleks.

Penerapan *game-based retrospective*: Starfish melibatkan penulisan 30 *sticky notes* selama 15 menit. Jumlah *sticky notes* di setiap bagian lingkaran Starfish bervariasi karena tidak ada batasan minimum atau maksimum. Tangkapan layar hasil *sprint retrospective* menggunakan metode ini dapat dilihat pada Gambar 3. Setelah menuliskan *sticky notes*, setiap anggota tim memilih dua *sticky notes* yang dianggap paling penting di setiap bagian dengan sesi voting selama tiga menit.



Gambar 3. Kumpulan *sticky notes* hasil *sprint retrospective* menggunakan game-based retrospective: Starfish

i. Bagian lingkaran *continue*

Area lingkaran "Continue" pada templat *game-based retrospective: Starfish* dirancang untuk menulis *sticky notes* mengenai pengalaman tim pengembang aplikasi Zikirin selama *sprint* terakhir. Tim menuliskan enam *sticky notes* pada bagian ini, namun, tidak semua anggota tim mencantumkan namanya pada *sticky notes* mereka sehingga tiga *sticky notes* tidak diketahui penulisnya. Salah satu *sticky notes* yang ditulis oleh peneliti, berisi tentang diskusi secara sinkron melalui Zoom, mendapatkan suara terbanyak pada sesi voting dengan lima suara. Tim mengungkapkan keinginan untuk terus melakukan sesi diskusi ini pada *sprint* berikutnya karena dianggap efektif dalam menangani masalah dan memudahkan komunikasi di antara anggota tim.

ii. Bagian lingkaran *more of*

Bagian "More of" pada templat *game-based retrospective: Starfish* dimaksudkan untuk mencatat pengalaman yang sudah dilakukan tim

pengembang aplikasi Zikirin selama *sprint* terakhir, tetapi perlu peningkatan intensitas dan kualitas di *sprint* berikutnya. Pada bagian "More of" ini, tim mengumpulkan tujuh *sticky notes*, meskipun tidak semua mencantumkan nama penulisnya. *Sticky note* yang mendapatkan *vote* terbanyak pada sesi voting di bagian "More of" adalah yang *sticky note* yang berisi permintaan salah satu anggota tim untuk lebih banyak melakukan *team bonding* dengan cara berdiskusi topik ringan di luar jam kerja. Topik ini mendapat lima suara oleh anggota tim pengembang aplikasi Zikirin yang lain.

iii. Bagian lingkaran *less of*

Bagian "Less of" pada *game-based retrospective: Starfish* digunakan untuk mencatat aktivitas yang perlu dikurangi intensitasnya di *sprint* berikutnya karena tidak memberikan manfaat yang signifikan untuk tim pengembang aplikasi Zikirin. Terdapat lima *sticky notes* pada bagian "Less of". *Sticky note* yang mendapatkan suara terbanyak pada bagian "Less of" adalah *sticky note* yang berisi aktivitas *screen sharing* permasalahan kecil yang membuat waktu diskusi tim terbuang secara percuma. Tim sepakat untuk mengurangi sesi *sharescreen* saat menyelesaikan masalah yang tidak terlalu penting selama diskusi di Zoom, karena dianggap menghabiskan waktu. Saran pengurangan aktivitas ini didukung oleh lima suara dari anggota tim pengembang aplikasi Zikirin lainnya. Anggota tim yang tidak memahami konteks masalah saat melihat sesi *sharescreen* menjadi bingung dan kurang produktif. Oleh karena itu, sesi *sharescreen* perlu dikurangi pada *sprint* berikutnya.

iv. Bagian lingkaran *stop*

Pada bagian "Stop" dalam *game-based retrospective: Starfish*, tim pengembang aplikasi Zikirin mencatat aktivitas-aktivitas yang sudah dilakukan pada *sprint* terakhir tetapi tidak perlu dilanjutkan pada *sprint* berikutnya. Ada empat *sticky notes* yang ditulis oleh tim pengembang aplikasi Zikirin, dan semua *sticky notes* pada bagian ini mencantumkan nama penulisnya. Isi *sticky note* yang mendapatkan suara terbanyak adalah permintaan peneliti untuk membuat *backlog* yang tidak terlalu besar supaya lebih mudah dikelola. Pada saat sesi voting, *sticky note* ini mendapatkan empat suara. Tim pengembang aplikasi Zikirin merasa bahwa *backlog* yang

ditulis selama beberapa sprint terakhir memiliki cakupan yang terlalu besar, sulit untuk dijaga selama *sprint* berlangsung, dan sulit dikerjakan oleh tim pengembang aplikasi Zikirin saat sprint berlangsung. Untuk *sprint* berikutnya, tim berharap tidak menulis *backlog* dengan cakupan besar lagi agar seluruh *backlog* yang ada di *sprint* selanjutnya mudah dijaga dan dipantau.

v. Bagian lingkaran *start*

Bagian "Start" pada *game-based retrospective*: Starfish berisi aktivitas-aktivitas baru yang ingin dicoba pada *sprint* berikutnya. Dalam bagian ini, tim pengembang aplikasi Zikirin menuliskan 8 *sticky notes*, dengan yang paling populer berisi saran untuk mencari perspektif desain lain yang ditulis oleh Rizki yang mendapatkan empat suara dari sesi voting. Tim setuju untuk mencari perspektif baru dalam desain dengan menyusun referensi-desain dari internet, karena merekrut desainer baru dianggap sulit. Hal ini diharapkan akan menghasilkan desain yang lebih menarik dan meningkatkan pengalaman pengguna aplikasi Zikirin.

KESIMPULAN

Tim pengembang aplikasi Zikirin telah menggunakan kerangka kerja Scrum sejak awal pengembangan aplikasi Zikirin. Namun, pada *sprint* pertama hingga *sprint* kelima, metode *sprint retrospective* tim pengembang aplikasi Zikirin hanya menggunakan Notion dengan dua pertanyaan "Apa yang sudah berjalan dengan baik?" dan "Apa yang harus di-improve?". Jawaban dari kedua pertanyaan ini tidak dapat mengetahui aktivitas-aktivitas apa saja yang sudah berjalan dengan baik dan apa saja yang perlu ditingkatkan pada sprint selanjutnya, serta tidak bisa menjabarkan aktivitas-aktivitas apa saja yang perlu dipertahankan, diperbanyak, dikurangi, dimulai, dan dihindari untuk sprint selanjutnya secara detail. Oleh karena itu, pada *sprint retrospective* di *sprint* keenam, peneliti berhasil menerapkan *game-based retrospective*: Starfish secara daring kepada tim pengembang aplikasi Zikirin untuk mengetahui secara lebih detail evaluasi-evaluasi yang perlu diperhatikan tim di *sprint* selanjutnya yang mencakup aktivitas-aktivitas yang perlu dipertahankan, diperbanyak, dikurangi, dimulai, dan dihindari.

UCAPAN TERIMA KASIH

Ucapan terima kasih ditujukan kepada semua pihak yang terlibat dalam pembuatan naskah ini dan pihak masyarakat. Ucapan terima kasih dapat pula ditujukan kepada penyedia dana atau sponsor.

DAFTAR PUSTAKA

- Cucolaş, A.-A., & Russo, D. (2023). The impact of working from home on the success of Scrum projects: A multi-method study. *Journal of Systems and Software*, 197, 111562. <https://doi.org/10.1016/j.jss.2022.111562>
- Fowler, F. M. (2019). *Navigating Hybrid Scrum Environments: Understanding the Essentials, Avoiding the Pitfalls*. Apress. <https://doi.org/10.1007/978-1-4842-4164-6>
- Mesquida, A.-L., Mas, A., Jovanović, M., & Radaković, N. (2016). Agile Retrospective Games for Different Team Development Phases. *Journal of Universal Computer Science*, 22(12). <https://doi.org/10.3217/JUCS-022-12-1489>
- Ozoliņš, P. (2018). Preparation and Facilitation of Retrospective Meeting in Scrum process. *Information Technology and Management Science*, 21, 60–63. <https://doi.org/10.7250/itms-2018-0009>
- Przybyłek, A., Albecka, M., Springer, O., & Kowalski, W. (2022). Game-based Sprint retrospectives: Multiple action research. *Empirical Software Engineering*, 27(1), 1. <https://doi.org/10.1007/s10664-021-10043-z>
- Przybyłek, A., & Kotecka, D. (2017). Making agile retrospectives more awesome. 1211–1216. <https://doi.org/10.15439/2017F423>
- Schwaber, K. (2004). *Agile project management with Scrum*. Microsoft Press.
- Schwaber, K., & Sutherland, J. (2020, November). *Scrum Guide Indonesian*. The Home of Scrum. Scrum.org
- Staffbase, S., & Civita, T. (2022). *Starfish Team Retro*. Figma. <https://www.figma.com/community/file/989686227639735026>



© 2023 by authors. Content on this article is licensed under a Creative Commons Attribution 4.0 International license. (<http://creativecommons.org/licenses/by/4.0/>).