

Perancangan *Dashboard* Sistem Informasi Dengan Metode *Scrum* Menggunakan *Azure Board*

Aria Bimantara^{1*}, Jaka Sutresna¹

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspiptek No. 46, Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: ¹aria.00a@gmail.com, ²dosen00833@unpam.ac.id

(* : coresponding author)

Abstrak— Permasalahan utama dari pelaksanaan proyek adalah sulitnya pengelolaan informasi yang disampaikan ke beberapa tim. Baik dari segi pengelolaan durasi pengerjaan proyek dan sumber daya. *Scrum* merupakan sebuah metode turunan dari *Agile Software Development* yang digunakan untuk menjalankan serta mengelola sebuah proyek pengembangan perangkat lunak. Metode *scrum* dapat mengorganisir sebuah tim dan harus memiliki komunikasi yang kuat antar *member* tim tersebut. Sebelum memulai *sprint*, tiap tim harus melakukan *meeting* singkat untuk perencanaan dan diakhiri dengan *review*. Menggunakan metode *scrum* diharapkan dapat mempersingkat waktu pengerjaan program karena disetiap tim mempunyai tugasnya masing-masing. Hasil yang diperoleh dapat mengatasi perubahan *requirements* pada saat fase pengembangan sistem, yang menghasilkan produk sesuai dengan keinginan pengguna karena mendapatkan *review* secara berulang (*iterative*).

Kata Kunci: *Scrum, Agile Software Development, Iterative*

Abstract— The main problem of project implementation is the difficulty of managing information conveyed to several teams. Both in terms of managing the duration of project work and resources. *Scrum* is a derivative method of *Agile Software Development* that is used to run and manage a software development project. The *scrum* method can organize a team and must have strong communication between team members. Before starting a *sprint*, each team must have a short meeting for planning and end with a review. Using the *scrum* method is expected to shorten the program work time because each team has its own tasks. The results obtained can overcome changing requirements during the system development phase, which produces products in accordance with user desires because they get repeated reviews (*iterative*).

Keywords: *Scrum, Agile Software Development, Iterative*

1. PENDAHULUAN

PT. Arranet Indonesia Sejahtera didirikan pada tahun 2016 sebagai perusahaan yang bergerak dibidang penyedia jasa dan teknologi sistem pembayaran *digital*, atau biasa disebut dengan *Payment Switching* atau *Middleware*. Dengan pilar produk utama arranet diantaranya *Biller Agregator*, *Octopus Middleware*, *PayAja Closed Loop Community* dan *Payment Advices*. PT. Arranet terintegrasi dengan *Azure DevOps* yang merupakan layanan kolaborasi dan *CI/CD* untuk membantu perencanaan, pengujian, hingga distribusi proyek yang kecil hingga kelas bisnis. *Azure DevOps* ini merupakan *manufacture* yang *didevelop* oleh *Microsoft*. Sehingga *Azure DevOps* sudah terintegrasi dengan produk-produk *Microsoft* yang lainnya.

Secara umum pengelolaan proyek di PT. Arranet dilakukan oleh 3 divisi utama. Yang pertama Divisi *Software Engineer* (SE), atau yang bisa disebut sebagai *programer*. Yang kedua Divisi *Quality Assurance* (QA), atau yang disebut sebagai *tester*. Yang ketiga Divisi *Customer Success* (CS), atau yang bisa disebut sebagai tim operasional. Dalam proses pengelolaan proyek di PT. Arranet, awal mula masih menggunakan papan manual berupa styrofoam dan sticky notes. Ketika proyek diterima oleh Koordinator Divisi *Software Engineering* (Koord SE), belum ada divisi *Project Management Office* (PMO). Detail dari sebuah proyek akan dituliskan di *sticky notes* dan ditempelkan di *styrofoam*. Proyek akan mulai dikerjakan oleh Divisi SE dan Divisi QA. Selama proses pengerjaan proyek, sticky notes akan terus berpindah ke kanan (ke kotak styrofoam yang lain) sesuai tahapannya. Hingga diakhiri proses Rollout atau Deployment, sticky notes akan dialihkan ke Divisi CS.

Seiring berkembangnya perusahaan, proyek mulai banyak yang masuk. Sehingga pengelolaan proyek menggunakan papan manual dirasa mulai tidak efektif. Mengakibatkan

kerangka kerja yang dilakukan tidak bisa dilakukan secara berulang (*iteratve*). Dan proses tracking sebuah proyek tidak terpantau aktifitasnya karena keterbatasan lebar *sticky notes* tersebut.

Maka dibutuhkan sebuah alat bantu virtual yang dapat membantu tim dalam berkomunikasi agar informasi yang diterima sesuai dengan kebutuhan proyek. Sinkronisasi pekerjaan serta visibilitas proyek yang dapat memberikan informasi akan kinerja tim selama proyek berlangsung. Serta dibutuhkan sebuah kerangka kerja yang mampu dilakukan secara berulang (*iterative*) hingga produk yang dikembangkan memenuhi kebutuhan sesuai dengan keinginan. Setiap iterasi akan melibatkan tim pada siklus pengembangan tersebut. Seperti analisis kebutuhan, perencanaan, desain, hingga implementasi testing. Dengan tujuan mampu beradaptasi pada setiap perubahan yang dapat terjadi sewaktu-waktu.

Berdasarkan masalah yang dipaparkan maka akan melakukan penelitian perancangan Dashboard Sistem Informasi Dengan Metode *Scrum* Menggunakan *Azure Board* (Studi Kasus : PT. Arranet Indonesia Sejahtera).

2. METODOLOGI PENELITIAN

2.1 Uraian Materi

2.1.2 Dashboard Sistem Informasi

Dashboard adalah aplikasi sistem informasi yang menyajikan informasi mengenai indikator utama dari aktifitas organisasi secara sekilas dalam layar tunggal. Pembuatan model memperhatikan 3 (tiga) aspek utama *dashboard* yaitu penyajian data/informasi, personalisasi, dan kolaborasi antar pengguna. Model yang dihasilkan digunakan untuk pengembangan dashboard bagi kebutuhan *monitoring* dan evaluasi kinerja. Tujuan utama *dashboard* adalah untuk membantu membuat keputusan yang tepat dan cepat berdasarkan dari data yang ada. *Monitoring* dan evaluasi mutlak dilakukan secara terus menerus oleh pengguna *dashboard* untuk memastikan bahwa proses bisnis yang dijalankannya dapat mencapai tujuan yang telah ditetapkan, melalui strategi pengelolaan yang tepat.

2.1.2 Metode Scrum

Scrum merupakan model baru untuk sebuah metode kerangka kerja dari metodologi *Agile*. *Scrum* diharapkan dapat meningkatkan kecepatan dan fleksibilitas dalam pengembangan perangkat lunak. *Scrum* merupakan metode *Agile* yang cukup *popular* dimana produktifitas menjadi lebih tinggi. Prinsip kerja dari *Scrum* adalah bekerja secara iteratif dan bertahap hingga mencapai waktu yang telah ditentukan, sehingga produk perangkat lunak yang dikembangkan dapat memenuhi kebutuhan yang diinginkan oleh konsumen. Metode *Scrum* dalam pengembangan perangkat lunak terbagi menjadi beberapa bagian *iteration* dimana setiap *iteration* disebut dengan *Sprint*.

Metode *Scrum* diawali dengan mengumpulkan seluruh kebutuhan pengguna, namun tidak menutup kemungkinan bahwa semua kebutuhan harus berasal dari pengguna disaat awal-awal *Sprint*. Pengguna dapat mengubah kebutuhan kapanpun ketika didalam proses pengembangan. Mereka bisa menghapus, memperbarui ataupun menambahkan sebuah fitur.

a. Pilar Scrum

1. **Adaptasi.** Bila variabel yang telah ditinjau ternyata akan menyebabkan hasil akhir yang tidak sesuai dengan harapan, maka perencanaan awal harus diubah.
2. **Transparansi.** Semua variabel yang perlu diketahui dibuat transparansi agar semua orang yang peduli dengan pengembangan perangkat lunak dapat melihatnya. Transparansi mengharuskan aspek-aspek tersebut didefinisikan dengan standar yang sama, sehingga semua orang yang terlibat memiliki pemahaman yang sama mengenai apa yang sedang ditinjau.
3. **Inspeksi.** Semua variabel yang dibuat transparan tersebut ditinjau setiap waktunya, peninjauan sehendaknya tidak dilakukan terlampaui sering karena dapat menyebabkan terhambatnya pekerjaan. Paling bermanfaat jika dilakukan secara rutin oleh peninjau yang kompeten pada saat pekerjaan berjalan.

b. Artefak Scrum

Artefak *Scrum* merepresentasikan pekerjaan atau nilai, bertujuan untuk memberikan ruang transparansi, dan kesempatan-kesempatan untuk meninjau ulang dan adaptasi akan perubahan yang telah akan terjadi. Artefak yang didefinisikan oleh *Scrum* secara khusus dirancang untuk meningkatkan transparansi dari setiap informasi yang ada antara tim pengembang, *product owner* dan *Scrum master*, dengan begitu semua pihak dapat memiliki pemahaman yang sama terhadap artefak.

1. Product Backlog

Merupakan daftar terurut semua fitur yang telah diketahui saat ini yang harus ada di dalam produk. *Product Backlog* adalah satu-satunya daftar sumber kebutuhan untuk setiap perubahan yang terjadi didalam produk. Salah satu cara mendapatkan *Product Backlog* berasal dari *User Story* yang telah dibuat sebelumnya. *Product Owner* bertanggung-jawab terhadap *Product Backlog*, termasuk isinya, ketersediaannya, dan urutannya. *Product Backlog* sendiri merupakan sentral dari *Scrum*.

2. Sprint Backlog

Sprint Backlog adalah sekumpulan *item Product Backlog* yang telah dipilih untuk dikerjakan di *Sprint*, juga di dalamnya rencana untuk mengembangkan potongan tambahan produk dan merealisasikan *Sprint Goal*. *Sprint Backlog* adalah perkiraan mengenai fungsionalitas apa yang akan tersedia di Inkremen selanjutnya dan pekerjaan yang perlu dikerjakan untuk menghantarkan fungsionalitas tersebut menjadi potongan tambahan produk yang “Selesai”.

3. Inkremen

Merupakan gabungan dari semua *item Product Backlog* yang diselesaikan pada *Sprint* berjalan dan nilai-nilai dari Inkremen *Sprint-Sprint* sebelumnya. Pada akhir *Sprint*, inkremen terbaru harus “Selesai”, yang artinya berada dalam kondisi yang berfungsi penuh dan memenuhi definisi “Selesai” yang dibuat oleh Tim *Scrum*. Terlepas apakah *Product Owner* akan merilis produknya, produk harus selalu berada dalam kondisi yang berfungsi penuh.

4. Definisi “Selesai”

Ketika sebuah *item Product Backlog* atau Inkremen dikatakan “Selesai”, setiap pihak harus mengerti dengan apa yang dimaksud dengan “Selesai”. Walaupun definisi ini berbeda-beda antar tim *Scrum*, sesama anggota tim harus memiliki pemahaman yang sama mengenai pekerjaan yang harus mereka selesaikan guna memastikan adanya transparansi. Ini adalah definisi selesai untuk Tim *Scrum* dan ini digunakan untuk memeriksa apakah pekerjaan untuk mengembangkan Inkremen dianggap selesai.

c. Tahapan Scrum

Tahapan *Scrum* dibuat untuk menciptakan kesinambungan dan mengurangi tahapan lain yang tidak tercantum di dalam *Scrum*. Tidak adanya pelaksanaan salah satu tahapan ini akan mengurangi transparansi dan menghilangkan kesempatan untuk meninjau dan membuat perubahan. Hal utama dalam tahapan *Scrum* biasa disebut dengan *Sprint*.

Sprint merupakan sebuah fase pengembangan perangkat lunak yang memiliki batasan waktu maksimal satu bulan, dengan durasi yang konsisten sepanjang proses pengembangan produk. *Sprint* baru, langsung dimulai ketika *Sprint* sebelumnya selesai. *Sprint* memuat dan terdiri dari *Sprint Planning*, *Daily Scrum*, *Sprint Review* dan *Sprint Retrospective*.

1. Sprint Planning

Pekerjaan yang dilakukan di dalam *Sprint*, direncanakan pada *Sprint Planning*. Perencanaan ini dibuat secara kolaboratif oleh seluruh anggota Tim *Scrum*. *Sprint Planning* dibatasi maksimum delapan jam untuk *Sprint* yang berdurasi satu bulan.

2. *Daily Scrum*

Merupakan kegiatan dengan batasan waktu maksimum selama 15 menit agar Tim Pengembang dapat mensinkronisasikan pekerjaan mereka dan membuat perencanaan untuk 24 jam kedepan.

3. *Sprint Review*

Diadakan di setiap akhir *Sprint* untuk meninjau Inkremen dan merubah *Product Backlog* bila diperlukan. Pada saat *Sprint Review*, Tim *Scrum* dan *stakeholder* berkolaborasi untuk membahas apa yang telah dikerjakan dalam *Sprint* yang baru usai. Berdasarkan hasil tersebut dan semua perubahan *Product Backlog* pada saat *Sprint*.

4. *Sprint Retrospective*

Merupakan kesempatan bagi Tim *Scrum* untuk meninjau dirinya sendiri dan membuat perencanaan mengenai peningkatan yang akan dilakukan di *Sprint* berikutnya. *Sprint Retrospective* dilangsungkan setelah *Sprint Review* selesai dan sebelum *Sprint Planning* berikutnya. Ini adalah tahapan dengan batasan waktu maksimum selama tiga jam untuk *Sprint* yang berdurasi satu bulan.

d. *Tim Scrum*

Tim *Scrum* tidak mencakup peran tradisional seperti *programmer*, *desainer*, *tester* atau analisa sistem. Tetapi para *stakeholder* yang bekerja sama dalam proyek untuk menyelesaikan pekerjaan mereka secara kolektif dan berkomitmen untuk menyelesaikannya. Terdiri atas *Product Owner*, Tim Pengembang dan *Scrum Master*.

1. *Scrum Master*

Fasilitator yang melayani tim tetapi bukanlah *project manager*. Memastikan fase metodologi *Scrum* dilakukan dengan baik, melindungi tim dan memastikan mereka tidak terlalu berkomitmen dengan yang dikerjakan karena perubahan dapat terjadi disetiap *Sprint*.

2. *Product Owner*

Pihak yang bertanggung jawab terhadap suksesnya pengembangan produk dan hasil kerja tim pengembang, biasanya adalah representasi dari konsumen. *Product Owner* satu-satunya orang yang bertanggung jawab mengelola *Product Backlog*.

3. *Tim Pengembang*

Sekelompok orang yang bertanggung jawab untuk menghasilkan *product* perangkat lunak. Dibentuk dan didukung oleh organisasi untuk mengatur dan mengelola pekerjaannya secara mandiri. Sinergi yang ada di dalam tim akan meningkatkan efisiensi dan efektifitas dari Tim Pengembang secara keseluruhan.

e. *Aplikasi Azure Boards*

Azure Boards merupakan aplikasi keluaran *Microsoft Corporation* yang memfokuskan pada pengelolaan proyek perangkat lunak. *Azure Boards* menyediakan tim pengembangan perangkat lunak dengan alat interaktif dan dapat dikonfigurasi sesuai kebutuhan guna mengelola proyek perangkat lunak. Menyediakan serangkaian kemampuan yang kaya termasuk dukungan asli untuk proses *Agile*, *Scrum*, dan *Kanban*, tampilan kalender, *dashboard* yang dapat dikonfigurasi, dan pelaporan terintegrasi. Alat-alat ini menskalakan seiring pertumbuhan bisnis. Berikut beberapa keunggulan aplikasi *azure boards*.

- Melacak Berbagai Jenis Pekerjaan
- Alat *Visual* Dan Interaktif
- Mengelola Prioritas
- *Support* Untuk Proses *Scrum*
- Terintegrasi Dengan *Github*
- *Dashboard* Yang Dapat Dikonfigurasi
- Mudah Disesuaikan.

3. ANALISA DAN PEMBAHASAN

3.1 Analisis Pengguna

Berikut penjelasan pengguna yang akan menggunakan aplikasi Azure Board di PT. Arranet Indonesia Sejahtera sesuai divisi:

- a. *PMO (Project Management Office)*. Adalah sebuah divisi atau departemen yang menentukan dan menjaga standar dalam manajemen proyek dalam PT. Arranet Indonesia Sejahtera.
- b. *SE (Software Engineering)*. Adalah sebuah divisi atau departemen yang bertugas melakukan analisa, membuat rekayasa, menyusun spesifikasi, mengimplementasikan dan memvalidasi suatu rancangan sistem perangkat lunak untuk menjawab suatu permasalahan. Divisi ini diisi oleh para *programmer*.
- c. *QA (Quality Assurance)*. Adalah sebuah divisi atau departemen yang bertanggung jawab untuk memastikan sebuah produk yang akan dilepas ke pasaran sudah memenuhi semua standar kualitas untuk setiap komponennya. Divisi ini diisi oleh para *tester*.
- d. *CS (Customer Success)*. Adalah sebuah divisi atau departemen yang berperan untuk memberikan pelayanan kepada pelanggan pada saat sebelum, selama, atau setelah pembelian produk atau layanan. Divisi ini diisi oleh tim operasional.

3.2 Analisis Kebutuhan Web

Berikut ini merupakan analisis kebutuhan pada *Dashboard* Sistem Informasi:

- a. Halaman *Login* untuk seluruh *user* mengakses Dashboard Sistem Informasi.
- b. Halaman *Overview Dashboard*, untuk melihat secara menyeluruh tampilan *graphic* dari data proyek.
- c. Halaman *Features Boards*, untuk menyimpan seluruh informasi proyek yang sedang dikerjakan. Dan menampilkan posisi proyek saat ini.
- d. Halaman *Features Backlog*, untuk mencatat daftar sumber kebutuhan untuk setiap perubahan yang terjadi didalam *features*. Dan menampilkan posisi *backlog* saat ini.

3.2 Analisis Kebutuhan Perangkat

Berikut penjelasan pengguna yang akan menggunakan aplikasi Azure Board di PT. Arranet Indonesia Sejahtera sesuai divisi

Berikut ini merupakan analisis kebutuhan pada *Dashboard* Sistem Informasi:

a. Perangkat Keras

Spesifikasi Laptop yang digunakan:

Tabel 1. Spesifikasi Perangkat Keras

SISTEM OPERASI	<i>Windows 8 / 8.1 / 10 / 11</i>
PROCESSOR	<i>Intel Core i5 / i7 / i7 (Extreme Edition) / i9</i>
RAM	<i>Minimum 4 GB</i>
HARDISK	<i>Minimum SSD 120 GB</i>

b. Perangkat Lunak

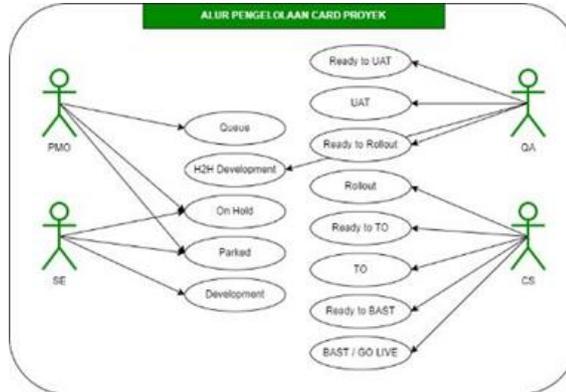
Daftar perangkat lunak yang dibutuhkan:

Tabel 2. Spesifikasi Perangkat Lunak

DASHBOARD	<i>Azure Boards</i>
BROWSER	<i>Google Chrome / Mozilla Firefox</i>

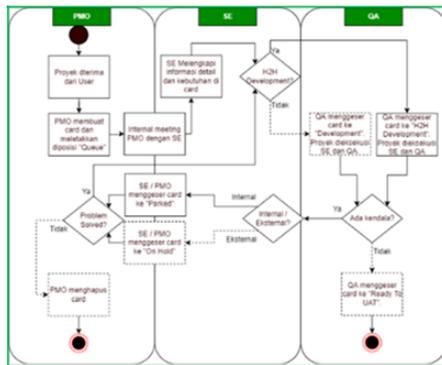
4. IMPLEMENTASI

4.1 Use Case Diagram

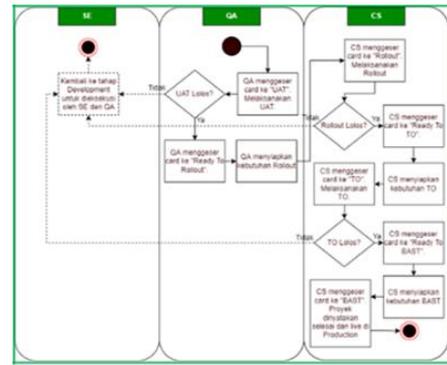


Gambar 1. Use Case Diagram untuk Alur Pengelolaan Card Project

4.2 Activity Diagram

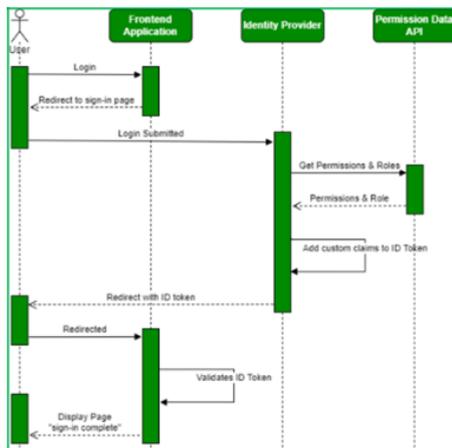


Gambar 2. Activity Diagram Development Process

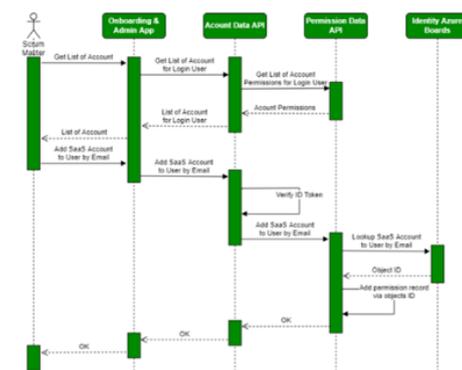


Gambar 3. Activity Diagram Deployment Process

4.3 Sequence Diagram

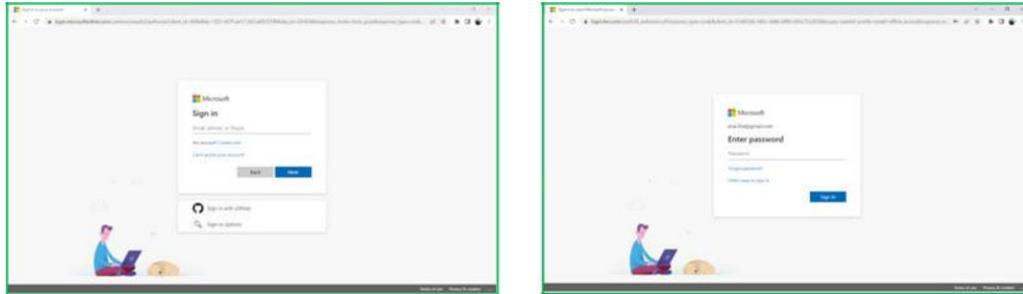


Gambar 4. Sequence Diagram Login



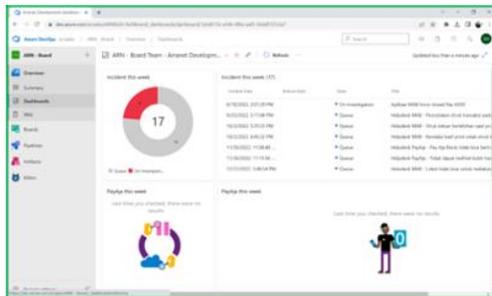
Gambar 5. Sequence Diagram New Account

4.4 Tampilan Login User

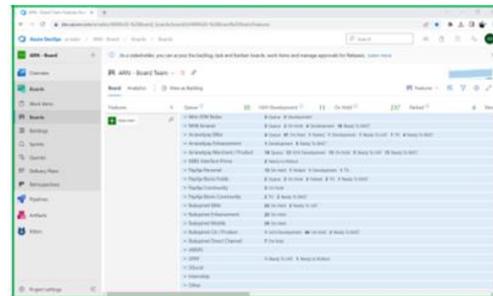


Gambar 7. Tampilan Login User

4.5 Tampilan Dashboard



Gambar 8. Tampilan Dashboard



Gambar 9. Tampilan Card Board

5. KESIMPULAN

Dari hasil analisis dan pembahasan, dapat disimpulkan sebagai berikut:

1. Dengan memilih tools *Azure Board*, mampu mengimplementasikan penggunaan *virtual sticky notes* untuk perancangan *dashboard* sistem informasi.
2. Dengan metode *Scrum*, mampu membuat kerangka kerja yang dapat dilakukan secara belurung (*iterative*).
3. Dengan penerapan sebuah “card” di *Azure Board* selaku pengganti *sticky notes*, mampu membuat banyak catatan sedetail-detailnya. Dan pergerakan *card* bisa terdeteksi sampai mana aktifitasnya.

REFERENCES

- Andry, J., Riwanto, R. E., Wijaya, R. L., & Prawoto, A. A. (2019). *DEVELOPMENT POINT OF SALES USING SCRUM FRAMEWORK*. Jakarta Utara: Universitas Bunda Mulia.
- Bahri, G. (2019). *PERANCANGAN DAN IMPLEMENTASI SISTEM MANAJEMEN PEMINJAMAN MOBIL DENGAN METODE SCRUM*. Batam: Universitas Internasional Batam.
- Center), B. (. (2022, December 16). *Mengenal Apa itu SaaS (Software as a Service) dan Contohnya*. Retrieved from Barantum Blog: <https://www.barantum.com/blog/saas-adalah/>
- Chandrasekara, C., & Herath, P. (2019). *Hands-on Azure Boards: Configuring and Customizing Process Workflows in Azure DevOps Services*. Apress.
- Elnour, A. H. (2020). Scrum as a Strategy for Learning Skills through Projects. *Didactic Proposal for its Implementation in the Education technology curriculum* , 40-54.
- IDwebhost. (2022, 12 02). *Apa Itu Scrum*. Retrieved from <https://idwebhost.com/blog/apa-itu-scrum/>



- Microsoft. (n.d.). *Microsoft*. Retrieved from Learn Microsoft Azure Architecture: <https://learn.microsoft.com/id-id/azure/architecture>
- Rashwan, M. F., & Saad, M. (2022). Agile Project Management Implications to Organization's Performance and Readiness for Digitalization. *Journal of Business and Management Sciences*, 233-236.
- Satriya, N. R. (2021). *IMPLEMENTASI METODE SCRUM DALAM PENGEMBANGAN PERANGKAT LUNAK DELIV*. Malang: Universitas Islam Negeri Maulana Malik Ibrahim.
- Supadnomo, W. A. (2022, January 18). *Mengenal Scrum untuk Implementasi Bisnis dan Inovasi yang Adaptif Terhadap Perubahan*. Retrieved from Medium.com: <https://medium.com/@wisnuariosupadnomo/mengenal-scrum-untuk-implementasi-bisnis-dan-inovasi-yang-adaptif-terhadap-perubahan-52c1a504d552>