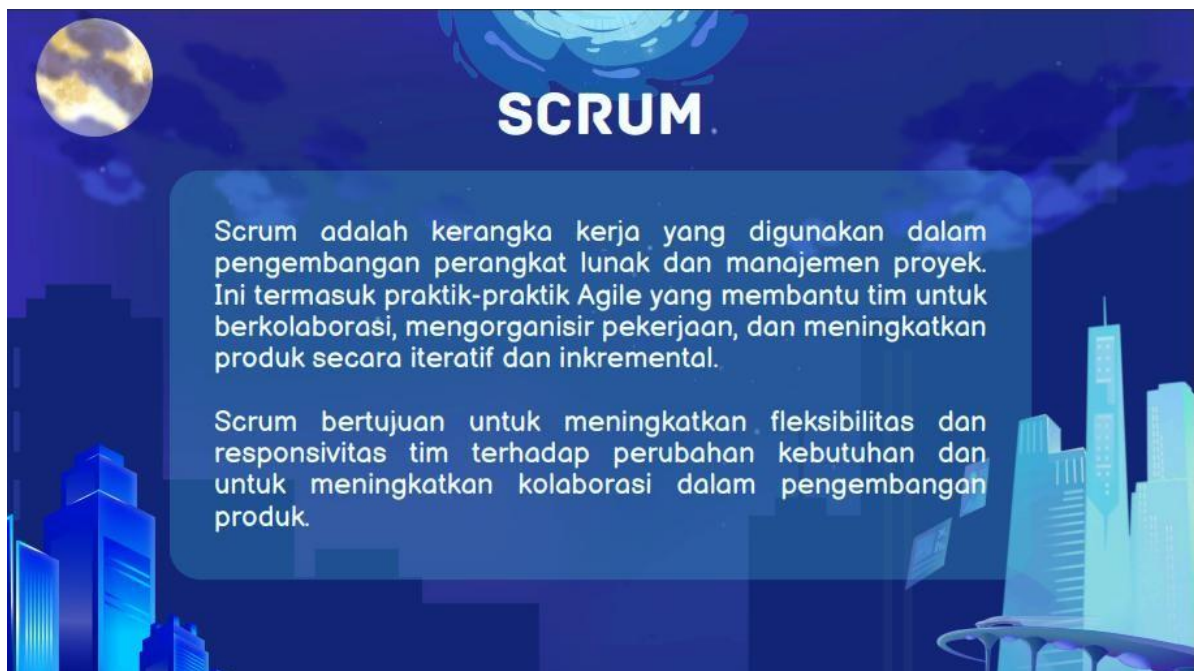


Tugas 4 Software Development & Management

Nama Anggota

- | | |
|-------------------|------------|
| 1. Eka Makhsusan | 2320506042 |
| 2. Restu Wibisono | 2340506061 |
| 3. Azqia Fattimah | 2340506067 |
| 4. Devan Putra | 2340506072 |

Tugas no 1



PERAN DALAM SCRUM

Scrum Master

Product Owner

Desainer

Tim Pengembang

Devan

Azqia

Eka

Restu

Menentukan visi proyek dan memprioritaskan Product Backlog

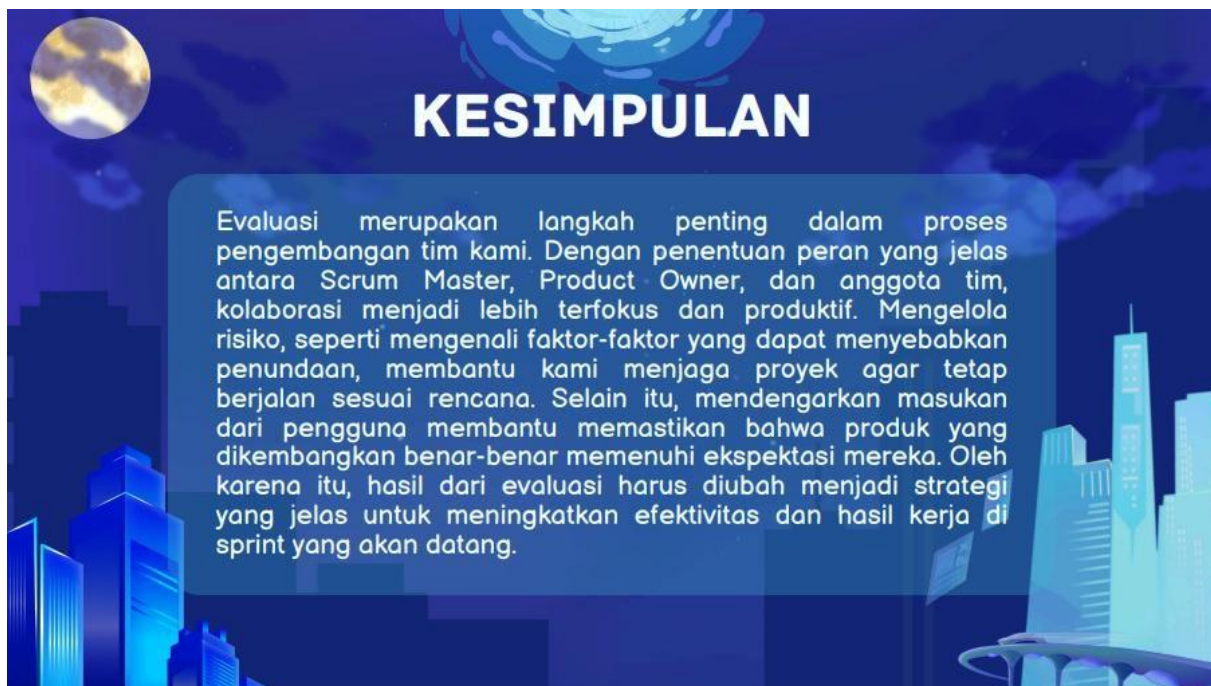
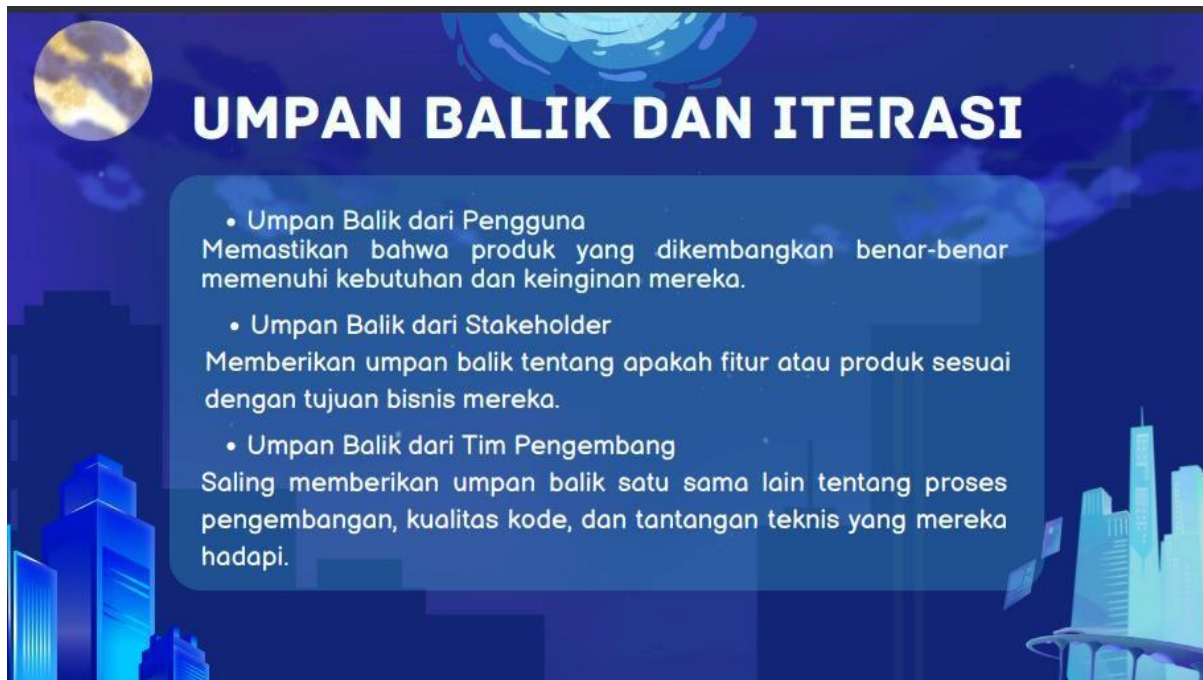
Memfasilitasi pertemuan dan mengatasi hambatan dalam tim

Membangun antarmuka pengguna (UI) aplikasi

Mengembangkan backend dan mengelola database

RENCANA KERJA

Backlog	To Do	In Progres	Done
Fitur Utama	Membuat Product Backlog	Membuat Desain UI	+ Add a card
Pengembangan Desain	Mendesain antarmuka UI	Membuat Kode Program	
Pengembangan Kode Program	Membuat Kode Program (Frontend dan Backend)	+ Add a card	
Analisis Bug dan Performa	+ Add a card		
Dokumentasi dan Dukungan			
+ Add a card			



Tugas no 2

Peran dalam Scrum

1. Scrum Master

Scrum Master berfungsi sebagai fasilitator bagi tim pengembang. Tugas utamanya meliputi:

- Menghilangkan Hambatan

Scrum Master bertanggung jawab untuk mengidentifikasi dan mengatasi

hambatan yang mungkin mengganggu kemajuan tim, seperti masalah teknis atau komunikasi yang tidak efektif

- Memfasilitasi Pertemuan

Dia memimpin pertemuan harian (daily stand-ups), retrospektif, dan perencanaan sprint, memastikan bahwa semua anggota tim terlibat dan diskusi tetap fokus.

- Mendukung Tim

Scrum Master juga berperan dalam mendukung tim dalam menerapkan prinsip-prinsip Agile dan Scrum, membantu mereka untuk bekerja secara mandiri dan efisien.

2. Product Owner

Product Owner adalah penghubung antara pemangku kepentingan dan tim pengembangan. Tanggung jawabnya meliputi:

- Mengelola Backlog Produk

Product Owner bertugas untuk membangun, memprioritaskan, dan memperbarui backlog produk. Ini termasuk merumuskan kebutuhan pengguna dalam bentuk user stories dan menetapkan kriteria penerimaan

- Menyampaikan Visi Produk

Dia harus mampu mengkomunikasikan visi produk dengan jelas kepada tim pengembang, memastikan bahwa semua orang memahami tujuan proyek.

- Berinteraksi dengan Pemangku Kepentingan

Product Owner sering berinteraksi dengan pemangku kepentingan untuk mendapatkan umpan balik dan memastikan bahwa produk yang dikembangkan sesuai dengan harapan mereka

3. Development Team

Tim pengembangan adalah kelompok profesional yang bertanggung jawab untuk mengembangkan produk. Tugas mereka meliputi:

- Pengembangan Fitur

Tim ini bertanggung jawab untuk merancang, mengembangkan, dan menguji fitur-fitur baru sesuai dengan backlog yang telah ditetapkan oleh Product Owner.

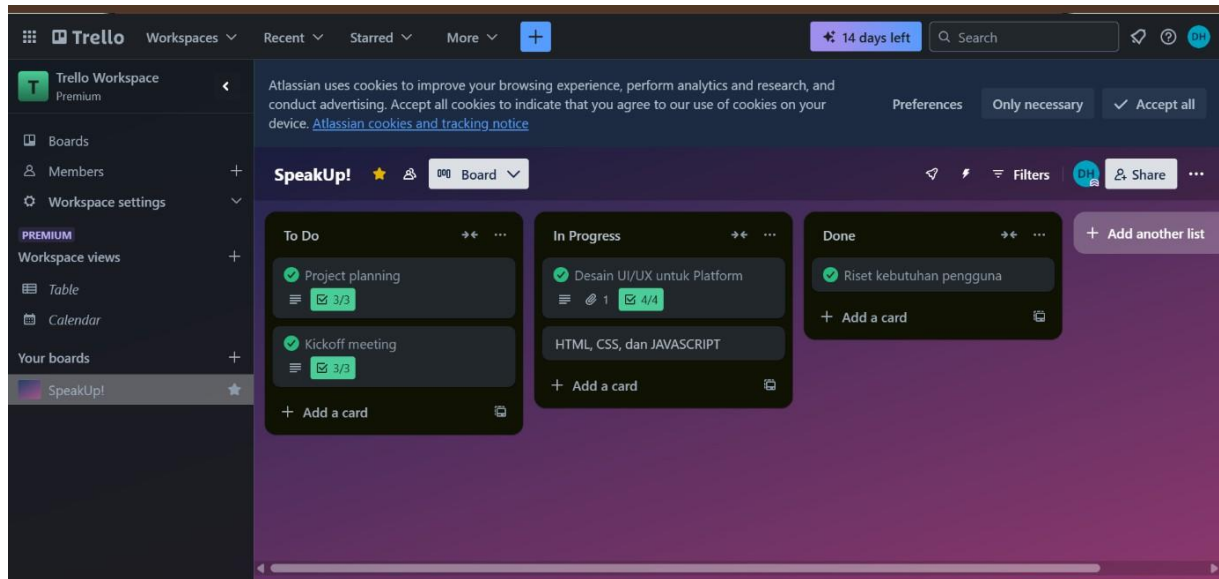
- Kolaborasi

Anggota tim harus bekerja sama secara efektif, berbagi pengetahuan, dan membantu satu sama lain dalam menyelesaikan tugas-tugas mereka.

- Evaluasi Kinerja

Tim juga terlibat dalam evaluasi kinerja mereka sendiri melalui retrospektif sprint, di mana mereka mendiskusikan apa yang berjalan baik dan apa yang perlu diperbaiki.

Tugas no 3



Tugas no 4

Prinsip simplicity (kesederhanaan) dalam Agile adalah tentang memaksimalkan jumlah pekerjaan yang tidak dilakukan atau dihilangkan. Tujuannya adalah untuk menghindari kompleksitas yang tidak perlu dalam pengembangan perangkat lunak, dengan fokus pada hal-hal yang benar-benar penting dan memberikan nilai bagi pelanggan. Prinsip ini menekankan pentingnya desain yang sederhana, kode yang bersih, dan proses yang efisien.

Berikut adalah beberapa cara bagaimana prinsip simplicity dapat diterapkan dalam pengembangan perangkat lunak:

1. Meminimalkan Fitur yang Tidak Perlu

Fokus pada pengembangan fitur-fitur inti yang benar-benar dibutuhkan oleh pengguna. Hindari menambahkan fitur-fitur yang mungkin berguna di masa depan tetapi tidak memberikan nilai langsung saat ini.

2. Desain yang Sederhana

Rancang sistem dan fitur dengan desain yang sesederhana mungkin. Gunakan pola desain yang sudah terbukti dan hindari arsitektur yang terlalu rumit.

3. Kode yang Bersih

Tulis kode yang mudah dibaca, dipahami, dan dipelihara. Gunakan konvensi penamaan yang jelas, hindari duplikasi kode, dan tulis komentar yang relevan.

4. Proses yang Efisien

Sederhanakan proses pengembangan dengan menghilangkan langkah-langkah yang tidak perlu. Otomatiskan tugas-tugas rutin seperti pengujian dan deployment.

5. Berfokus pada Nilai

Prioritaskan pekerjaan berdasarkan nilai yang diberikan kepada pelanggan. Mulailah dengan fitur-fitur yang paling penting dan berikan dampak terbesar.

6. Penggunaan tools yang tepat guna

Pilih tools atau alat yang tepat dan mudah digunakan untuk mempermudah pekerjaan tim. Hindari penggunaan tools yang terlalu kompleks atau memiliki fitur yang tidak diperlukan.

7. Komunikasi yang Efektif

Jaga agar komunikasi tetap sederhana dan langsung. Hindari penggunaan jargon teknis yang berlebihan dan pastikan semua anggota tim memahami tujuan dan prioritas proyek.

8. Umpan Balik yang Cepat

Dapatkan umpan balik dari pengguna atau pemangku kepentingan sesegera mungkin. Gunakan umpan balik ini untuk menyempurnakan produk dan memastikan bahwa produk memenuhi kebutuhan pengguna.

9. Iterasi dan Refaktor

Kembangkan perangkat lunak secara bertahap dan lakukan refactoring secara teratur. Refactoring adalah proses membersihkan dan menyederhanakan kode tanpa mengubah perilakunya.

10. Dokumentasi yang Ringkas

Buat dokumentasi yang ringkas dan relevan. Hindari membuat dokumentasi yang terlalu panjang atau detail, tetapi pastikan dokumentasi mencakup informasi penting tentang cara menggunakan dan memelihara perangkat lunak.

11. Fokus pada Tujuan

Ingatlah tujuan utama dari proyek dan hindari terjebak dalam detail-detail kecil yang tidak relevan. Pastikan semua keputusan dan tindakan selaras dengan tujuan proyek.

12. Berani Menghilangkan

Jangan takut untuk menghilangkan fitur atau komponen yang tidak lagi memberikan nilai. Terkadang, menghilangkan sesuatu dapat membuat sistem menjadi lebih sederhana dan mudah dipelihara.