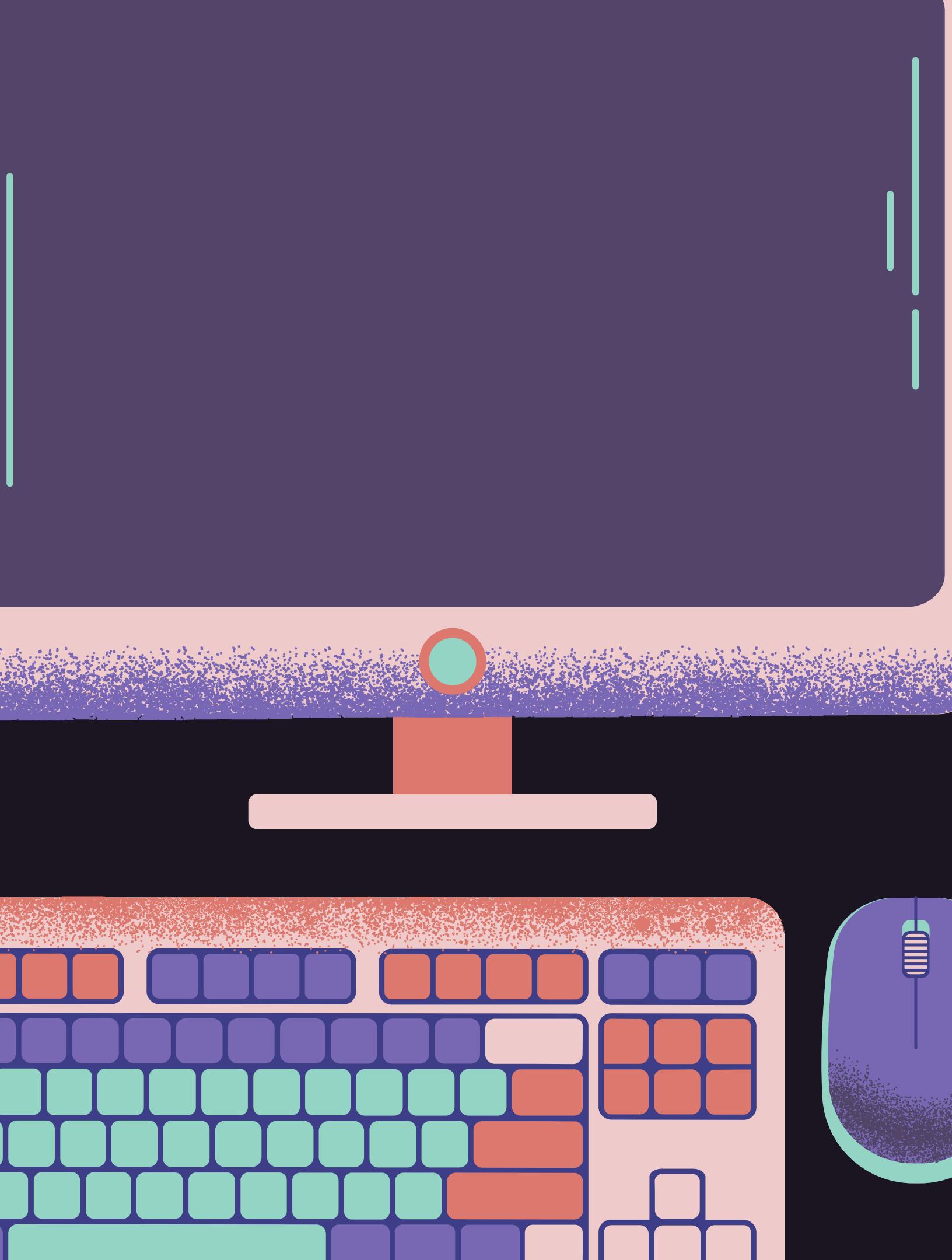


RPL0

UNIFIED SOFTWARE DEVELOPMENT PROCESS (USDP)

PRODI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA



ANGGOTA

NAYLA ANANDA

13020230112

RASMAWATI

13020230118

NUR AQIDAH SAFANIKIAH

13020230121

AQILAH JAMELLYA WULANDARI PUTRISYAH

13020230155

MUTIAH SHAULATIYA RISWAN

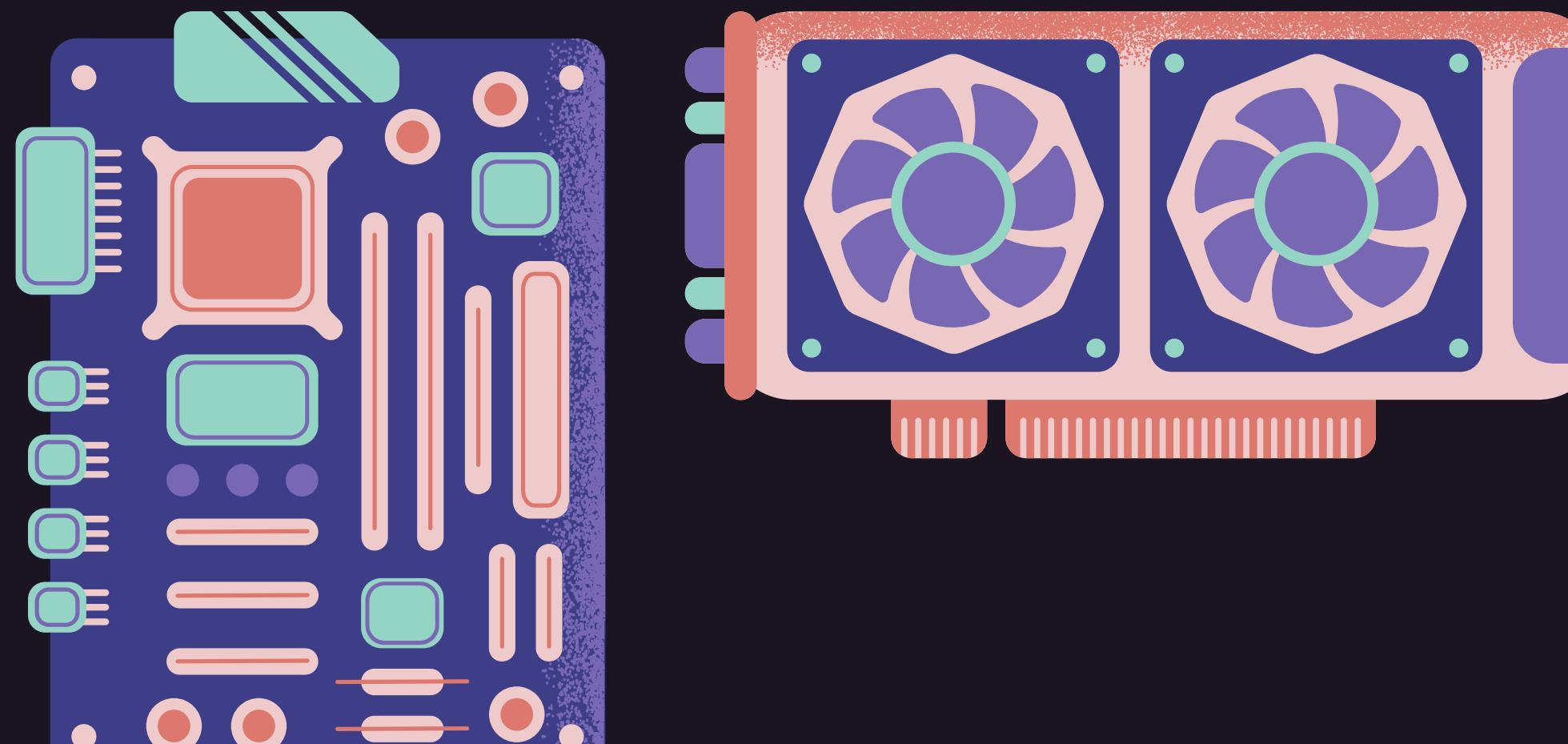
13020230173



SEJARAH

Unified Software Development Process (USDP) adalah metodologi pengembangan perangkat lunak yang iteratif, incremental, dan berorientasi objek, dikembangkan oleh Grady Booch, Ivar Jacobson, dan James Rumbaugh sebagai dasar dari Rational Unified Process (RUP). USDP menggabungkan pemodelan visual dengan UML, berfokus pada kebutuhan pengguna dan arsitektur perangkat lunak, serta mendukung penggunaan kembali komponen untuk efisiensi.

Metodologi ini fleksibel dan adaptif terhadap perubahan selama siklus pengembangan, meningkatkan keterlibatan pemangku kepentingan, dan cocok untuk proyek perangkat lunak besar dan kompleks.



FASE UTAMA USDP

Fase Inception

merupakan tahap permulaan di mana pengembang perangkat lunak melakukan interaksi dengan customer untuk mengidentifikasi kebutuhan-kebutuhan sistem yang hendak dibuat.

Fase Elaboration

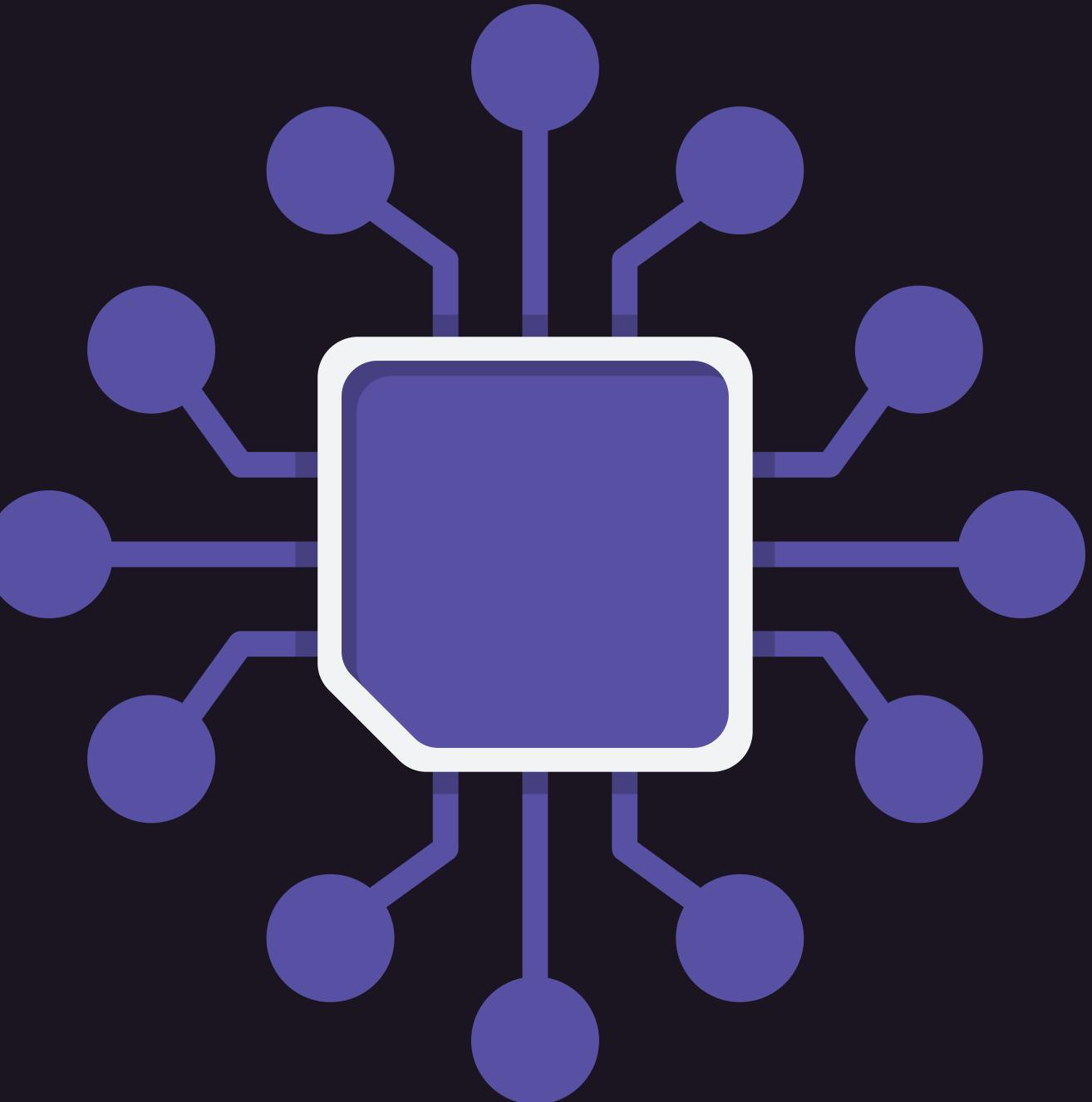
digunakan untuk mematangkan konsep-konsep yang sudah terbentuk di fase Inception.

Fase Construction

merupakan fase coding, di mana pengembang perangkat lunak mulai melakukan pembuatan sistem secara nyata.

Fase Transition

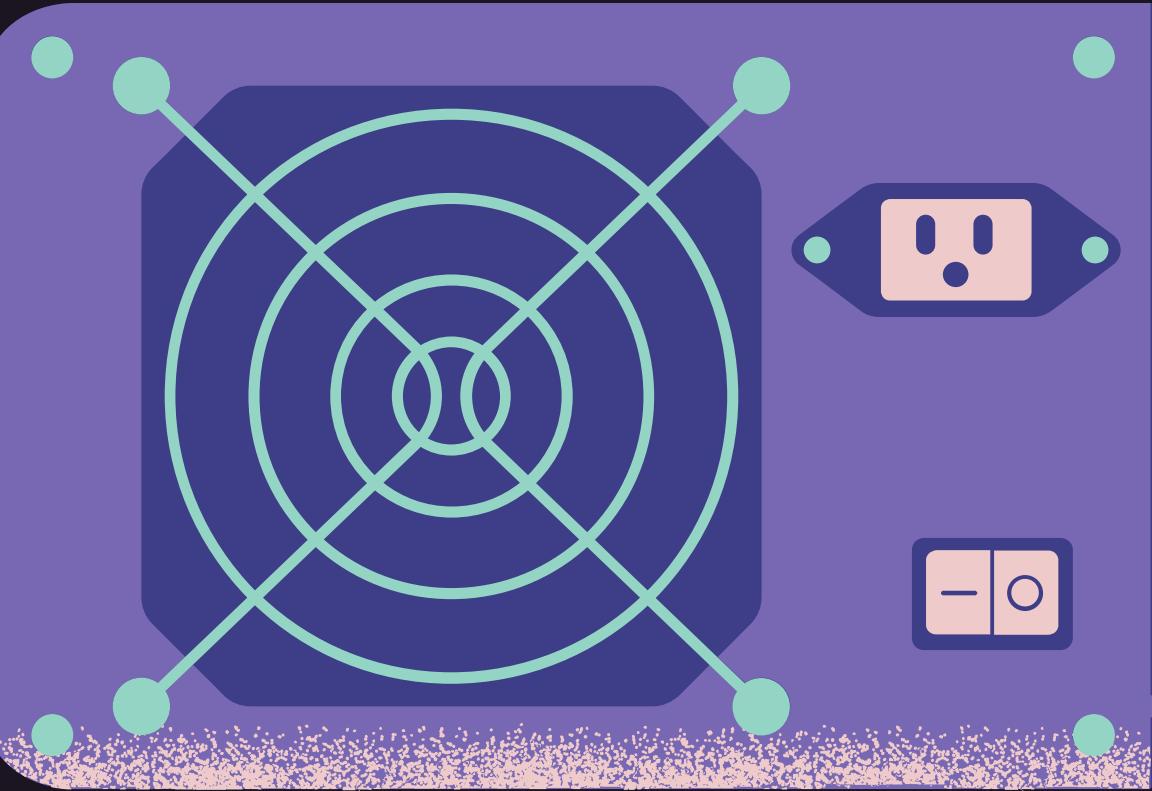
merupakan tahap untuk mematangkan produk akhir yang sudah jadi.



KARAKTERISTIK MODEL

Use-Case Driven (Digerakkan oleh Kasus Penggunaan)

Use case menggambarkan fungsionalitas sistem dari sudut pandang pengguna dalam format yang mudah dipahami.

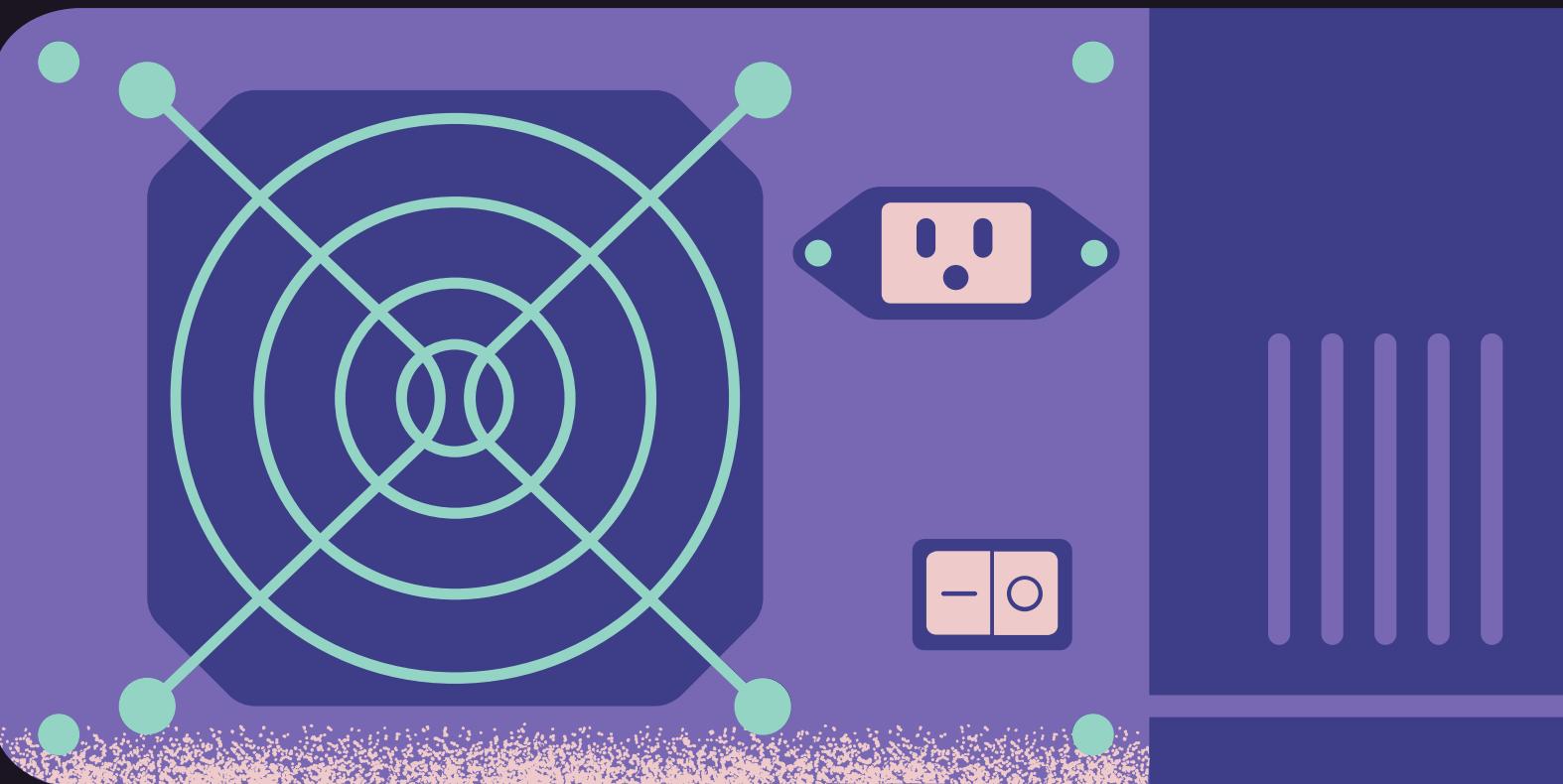


Architecture-Centric (Berpusat pada Arsitektur)

USDP mengutamakan arsitektur berbasis komponen untuk membangun sistem kompleks secara terstruktur dan fleksibel.

Iterative dan Incremental (Berulang dan Bertahap)

Pengembangan perangkat lunak dengan USDP dilakukan secara iteratif (berulang) dan incremental (bertahap). Prosesnya dibagi menjadi beberapa fase, dan di dalam setiap fase terdapat beberapa iterasi.



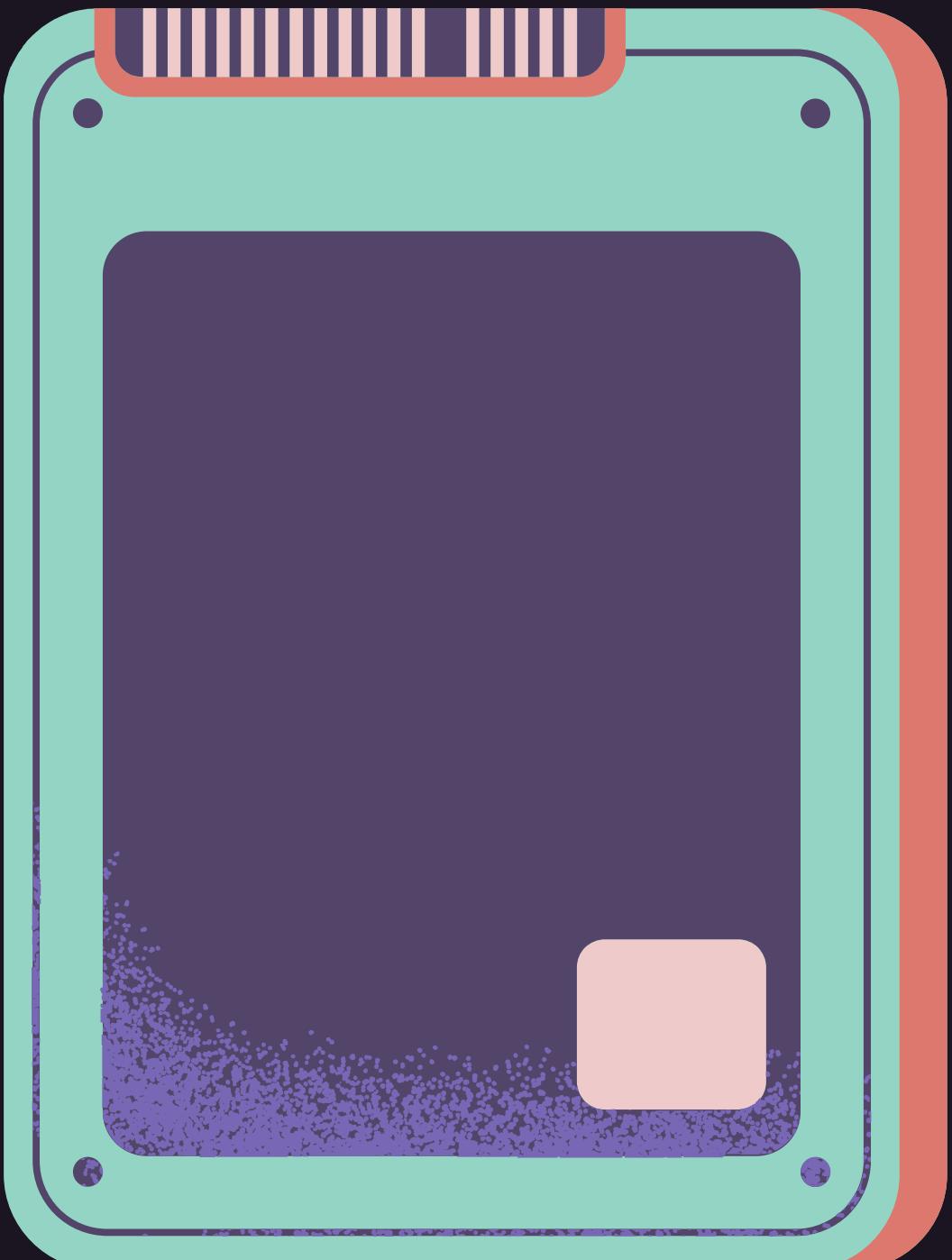
KELEBIHAN DAN KELEMAHAN

KELEBIHAN

- Mampu mengidentifikasi dan menangani risiko proyek secara efektif melalui manajemen permintaan dan review ketat.
- Scalable untuk tim dan proyek besar maupun kecil, fleksibel untuk single developer atau tim besar.
- Review berkala meningkatkan fokus dan transparansi proyek.
- Mendukung perubahan untuk peningkatan prototipe sehingga menghasilkan sistem yang dapat diterima.
- Mengantisipasi kebutuhan sistem yang kurang detail pada tahap awal melalui pengujian di akhir setiap fase.

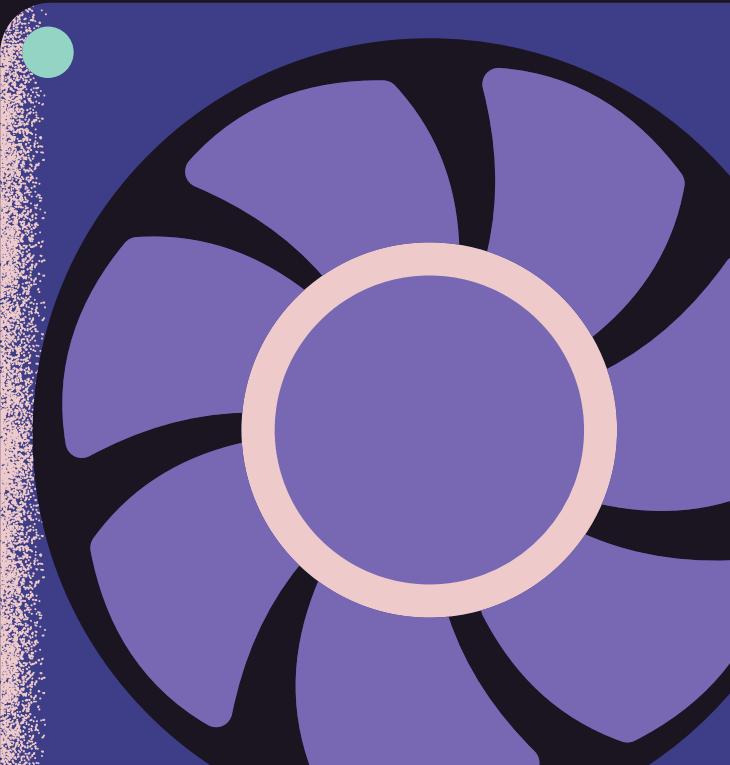
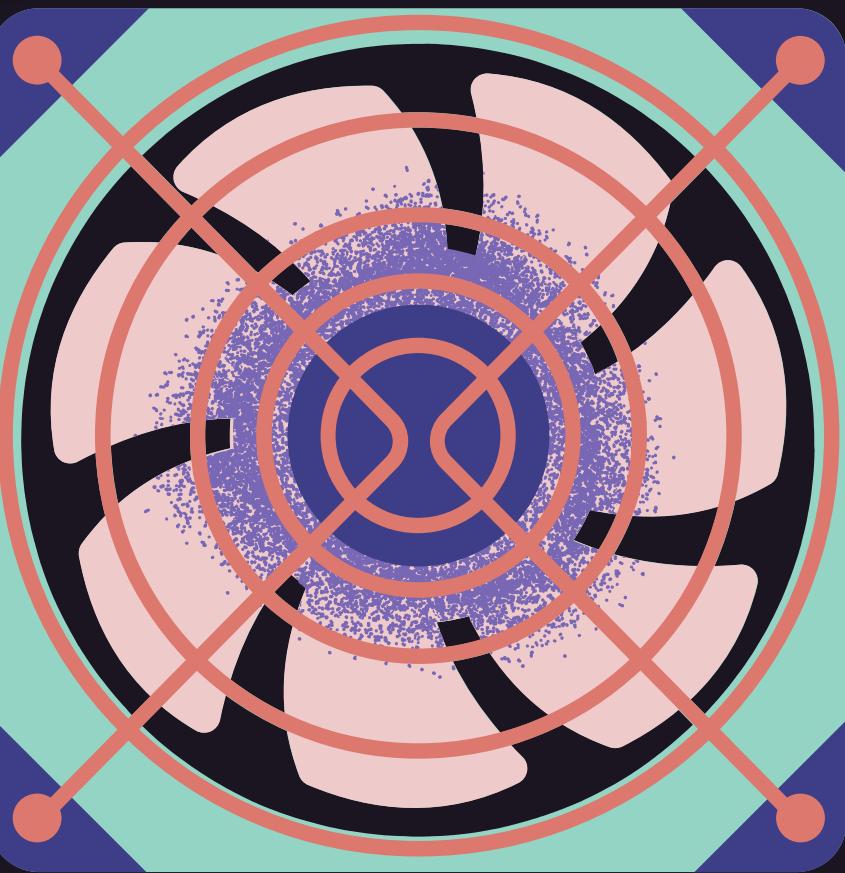
KEKURANGAN

- Proses pengembangan yang kompleks memerlukan keterampilan mendalam dari tim.
- Pengujian komponen yang berkelanjutan meningkatkan kompleksitas dan potensi masalah.
- Membutuhkan perencanaan dan sumber daya yang cermat untuk mengatasi tantangan pengujian.



PERBEDAAN USDP DENGAN METODE LAINNYA

USDP menggunakan pendekatan iteratif dengan dokumentasi lengkap dan manajemen risiko terstruktur, berbeda dari waterfall yang linear, Agile yang lebih fleksibel, XP yang fokus coding intensif, dan Scrum yang berbasis sprint pendek tanpa proses risiko khusus.



ALAT BANTU

USDP menggunakan UML untuk pemodelan sistem yang lengkap dan detail, lebih unggul dari Waterfall, Agile, dan Scrum dalam menangani kompleksitas dan manajemen risiko. Tools modern seperti IBM Rational Rose dan Eclipse mendukung otomatisasi dan efisiensi pengembangan dengan USDP.



KESIMPULAN

USDP adalah metodologi iteratif berbasis komponen yang menekankan arsitektur dan pemodelan lengkap dengan UML, cocok untuk proyek besar dan kompleks. Dibandingkan metode lain, USDP lebih terstruktur dan fokus pada manajemen risiko, meski memerlukan keahlian tinggi, dapat didukung oleh tools modern untuk efisiensi.

