B. Praktik Komputasi dengan Koding Blok

Setelah menyusun algoritma dalam bentuk diagram alur, selanjutnya diperlukan proses untuk mengubah algoritma tersebut menjadi sebuah kode program. Selain menggunakan pemrograman skrip, seperti Python, Java, dan C terdapat metode pemrograman lain, yaitu dengan menggunakan koding blok. Koding blok menggunakan representasi visual yang merepresentasikan perintah-perintah. Perhatikan penjelasan mengenai materi koding blok berikut.

1. Memahami Komputasi dengan Koding Blok

Koding blok merupakan solusi yang efektif untuk pemula dalam dunia pemrograman komputer. Koding blok menawarkan visual antarmuka yang mudah dipahami. Peserta didik dapat fokus pada ide kreatifnya menggunakan blok kode tanpa harus terjebak dalam penulisan kode yang rumit. Hal ini memungkinkan peserta didik untuk merasakan keberhasilan lebih cepat saat menciptakan program pertamanya sehingga meningkatkan motivasi dan minat dalam belajar koding.

Selain itu, koding blok juga mengajarkan keterampilan berpikir komputasional. Peserta didik belajar tentang urutan logis dan interaksi antarelemen dalam program yang membantunya memahami struktur serta aliran eksekusi perintah. Keterampilan ini tidak hanya bermanfaat pada pemrograman, tetapi juga untuk memecahkan masalah dan mengembangkan pola pikir analitis yang dapat diterapkan dalam berbagai aspek kehidupan.

Pendekatan ini telah mendapatkan pengakuan positif dari pendidik yang melihatnya sebagai fondasi kuat untuk transisi ke bahasa pemrograman yang lebih kompleks.

2. Jenis-Jenis Media Belajar Koding Blok

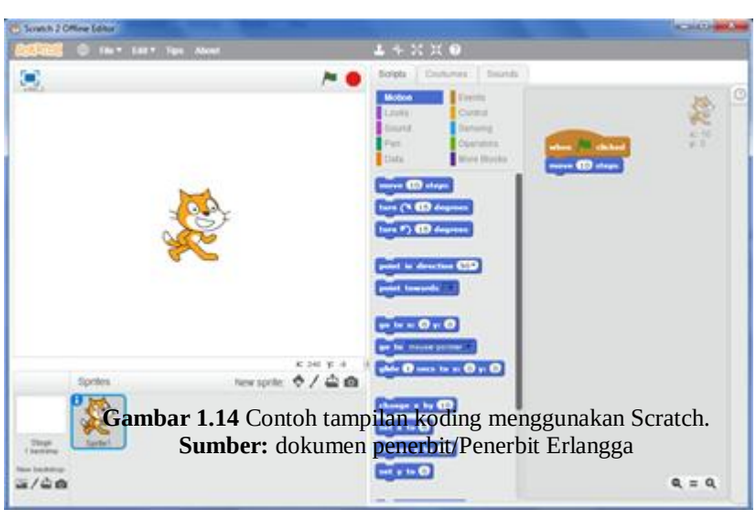
Pembelajaran yang menggunakan media belajar berbasis gim edukasi dengan koding blok dapat meningkatkan kemampuan dalam memecahkan masalah, rasa bahagia, dan percaya diri peserta didik **(Chiang & Qin, 2018)**.

Mengintegrasikan permainan digital ke dalam pendidikan khususnya melalui penggunaan mainan koding blok, terbukti menumbuhkan keterampilan kritis dalam mata pelajaran STEM (*Science*, *Technology*, *Engineering*, *and Mathematics*), menumbuhkan rasa ingin tahu, mendorong pertanyaan, serta eksplorasi konseptual **(Kamola et al., 2024)**.

Berikut adalah contoh media belajarnya.

a. Scratch

Penerapan koding blok dapat dilakukan dengan bantuan Scratch. Scratch tidak hanya meningkatkan minat peserta didik dalam belajar, tetapi juga mengembangkan keterampilan kognitif, kemampuan koding, serta kompetensi digitalnya sehingga pembelajaran menjadi lebih menyenangkan dan bermakna. Pendekatan ini mendorong partisipasi aktif dan kolaborasi, keterampilan memecahkan masalah, mengembangkan kompetensi digital, serta mendorong pembelajaran mandiri dan eksploratif. Secara keseluruhan, mengintegrasikan Scratch ke dalam pendidikan dan pembelajaran memiliki potensi besar untuk meningkatkan keterampilan kognitif, sosial, serta digital di antara peserta didik **(Rahmadika et al., 2024).**



Gambar 1.14 Contoh tampilan koding menggunakan Scratch.
Sumber: dokumen penerbit/Penerbit Erlangga

b. Blockly Games

Melatih kemampuan berpikir komputasional dapat dilakukan melalui praktik menggunakan aplikasi berbasis permainan edukatif. Salah satu aplikasi yang menawarkan permainan edukasi interaktif adalah Blockly Games. Aplikasi ini dirancang membantu pengguna memahami konsep logika serta pemrograman dasar secara menyenangkan dan bertahap. Penggunaan aplikasi ini mendorong pengembangan keterampilan berpikir sistematis, kritis, dan kreatif.

(a) Tantangan
(b) Koding visual blok
(c) Koding skrip yang benar

C. Praktik Problem Solving

Problem solving adalah kemampuan seseorang untuk dapat menyelesaikan masalah dengan cara berusaha mendapatkan jawaban sebagai jalan keluar. Kemampuan *problem solving* jarang terjadi begitu saja dari dalam diri seseorang tanpa adanya kesadaran untuk mempelajari dan melatih diri agar terbiasa melakukan kemampuan *problem solving*. Berikut dijelaskan lebih rinci mengenai praktik *problem solving*.

1. Memahami Masalah dan Kepekaan Sosial

Salah satu contoh masalah sosial yang terjadi di masyarakat adalah perselisihan antara tukang ojek konvensional dengan ojek online dalam mendapatkan penumpang. Tukang ojek sering menunggu di pangkalannya, tetapi jarang sekali mendapatkan pelanggan. Salah satu penyebabnya adalah tarif yang ditetapkan terlalu tinggi. Seiring perkembangan waktu, aplikasi ojek online mulai berkembang dan diterima secara luas di seluruh Indonesia. Ojek konvensional, kini banyak yang menjadi driver ojek online. Penggunaan ojek online sangat mudah, hanya dengan mengandalkan aplikasi. Penumpang dapat melihat tarif sebelum memesan sehingga tidak merasa terbebani. Berdasarkan permasalahan tersebut, lahirlah sebuah aplikasi dari kepedulian sosial dan diikuti dengan pengembangan teknologi yang akhirnya berhasil menghadirkan perubahan besar bagi banyak orang.

Pemahaman terhadap teknologi ini sebaiknya juga ditanamkan sejak dini melalui dunia pendidikan. Salah satu upaya yang relevan adalah penerapan pembelajaran kolaboratif dan tematik, seperti kolaborasi antara mata pelajaran Informatika atau Koding dan Matematika yang mengambil tema pemecahan masalah dengan pendekatan koding. Pelaksanaan pembelajaran dapat dilakukan di luar sekolah dengan melihat langsung permasalahan-permasalahan yang ada di lingkungan sekitar. Data hasil pengamatan diolah di lab komputer berbasis solusi. Hasil pembelajaran ini sangat luar biasa dan telah dilakukan serta banyak diteliti, salah satunya kolaborasi pelajaran matematika dengan pengodean (koding) **(Tshesane & Venkat, 2023)**.

Keterampilan digital, terutama koding menjadi sangat penting di dunia yang terus berkembang. Koding berfungsi sebagai dasar dari teknologi kini dan masa depan sehingga memungkinkan individu untuk menciptakan program, aplikasi, serta situs web yang mendukung kehidupan digital. Meningkatnya permintaan akan ahli koding membuat keterampilan ini menjadi kunci penting untuk berbagai peluang karier di industri teknologi, mulai dari pengembang perangkat lunak hingga insinyur kecerdasan artifisial (KA).

2. Menerapkan Berpikir Komputasional dalam Pemecahan Masalah

Berpikir komputasional merupakan pendekatan sistematis dalam menyelesaikan masalah. Salah satu penerapannya adalah melalui pembuatan program sebagai solusi dari permasalahan tertentu. Agar dapat membuat program yang efektif dan mendukung tugas manusia, terdapat tiga langkah yang dapat dilakukan dalam proses pemrograman, yaitu sebagai berikut.

a. Menganalisis permasalahan (*analyzing*)

Programer harus menganalisis kebutuhan atau kondisi yang ada untuk merumuskan definisi masalah yang perlu dipecahkan dengan program. Pada tahap ini, kemampuan berpikir komputasional digunakan untuk menemukan abstraksi dari masalah yang akan diselesaikan. Masalah yang kompleks dapat dibagi menjadi beberapa submasalah yang lebih kecil. Tahap ini menghasilkan pernyataan masalah (*problem statement*) yang mendeskripsikan masukan (*input*), keluaran (*output*), dan batasan (*constraints*) dari program yang akan dikembangkan.

b. Mendesain solusi (*problem solving*)

Berdasarkan permasalahan yang ada, programer merancang strategi untuk menghasilkan keluaran sesuai dengan masukan yang diterima. Programer bisa memanfaatkan solusi atau bagian dari solusi yang sudah ada dari masalah sebelumnya, tanpa perlu merancang solusi dari awal. Jika telah ditemukan masalah umum, programer bisa memodifikasi algoritma generik agar sesuai dengan konteks masalah yang dihadapi. Proses ini akan menghasilkan algoritma dalam bentuk narasi, *pseudocode*, atau diagram alur.

c. Menguji program (*testing*)

Programer perlu mengevaluasi algoritma yang telah dikembangkan untuk memastikan kepatuhannya terhadap batasan yang ada. Contohnya perlu dipastikan program dapat beroperasi dengan cepat (kurang dari satu detik) saat diberikan masukan berukuran besar karena waktu tunggu yang lama dapat mengganggu kenyamanan pengguna.

Setelah program berhasil dijalankan, langkah selanjutnya adalah mengujinya untuk memastikan program berfungsi dengan baik dan memenuhi batasan yang telah ditetapkan. Pengujian dapat dilakukan dengan berbagai metode. Salah satu cara untuk menguji program adalah menggunakan kasus uji yang dirancang untuk mewakili semua kemungkinan masukan yang dapat diterima oleh program.

3. Metode Problem Solving

Mempelajari ilmu progrwmming adalah salah satu kegiatan yang membantu untuk mengenal problem solving serta mempratikkannya Steve Jobs pernah mengatakan bahwa setiap orang perlu untuk mempelajari program computer karena dapat mengajarkan untuk berpikir. Kegiatan membmembuat program membantu anda untuk berpikir secara efektif dan efisien dalam menyelesaikan masalah. Tiap tahap dalam *problem solving* adalah langkah yang sangat perlu diperhatikan dengan baik oleh programmer berikut adalah metode yang dapat dilakukan dalam berpikir problem solving seperti programer.

a. Memahami permasaalahan yang terjadi

Memahami masalah merupakan tahapan pertama sebelum menyelesaikan masalah. sama seperti programmer, ketika dihadapkan pada masalah, anda perlu memahami permasalahan yang sebenarnya sedang dihadapi di hadapi. Memahami masalah dapat dilakukan dengan berbagia cara. Programmer dapat memahami masalahnya dengan menggambar diagram alur menurut Richard Feynman, saat dapap menjelaskan masalah dengan sederhana berarti Anda sudah memahami masalah yang sedang dihadapi dengan baik.

b. Membuat rencana

Ketika menyelesaikan masalah. Para programmer membuat perencanaan untuk menimalkanadanya resiko yang terjadi saat suatu cara dilakukan. Hal tersebut dapat anda tiru dengan membuat rencana dan menganalisis risiko yang mungkin dapat terjadi jika melakukan rencana tersebut. Anda dapat melakukan cara yang memiliki resiko tinggi dan selesai dalam waktu yang singkat atau melakukan cara dengan resiko yang rendah, tetapi waktu penyelesaiannya sedikit lebih lama. Rencana dalam *problem solving* yang dibuat oleh programer digunakan sebagai pegangan agar tidak salah dalam menentukan solusi dari suati permasalahan.

c. Melakukan pembagaian masalah

Tidak semua masalah yang dihadapi adalah masalah yang besar dan berat, ada juga masalah kecil yang dapat diselesaikan dengan cepat. Ada juga masalah yang harus diselesaikan saat itu juga atau bahkan masalah yang dapat diselesaikan di akhir. Para programer membagi masalah menjadi submasalah, kemudian menyelesaikan masalah tersebut satu per satu. Anda dapat melakukan hal yang sama karena hal ini dapat membuat masalah terlihat lebih detail dan terselesaikan dengan efektif

d. Penyelesaian

Lakukan semua rencana penyelesaian di submasalah yang sudah dibuat sebelumnya. Jangan lupa memperhatikan prioritas masalah dan risiko yang akan dihadapi sebelum mengambil langkah penyelesaian. Berhati-hati dalam mengambil langkah adalah bagian dari penyelesaian masalah karena akan memperkecil risiko membuat masalah baru.

e. Berlatih

Mempelajari *skills* yang dimiliki programer tentunya tidak dapat dilakukan dalam jangka waktu yang singkat. Programer berlatih, belajar terus menerus, dan membiasakan diri dengan kebiasaan-kebiasaan dari berpikir *problem solving*. Sama seperti programer yang terus belajar, Anda juga perlu terus berlatih untuk terbiasa dengan pola pikir dan perilaku seperti ini. Agar ketika dihadapkan dengan masalah, Anda akan memberikan refleks untuk melakukan metode-metode berpikir dalam *problem solving*. Anda dapat menggunakan aplikasi Leetcode dan Hackerrank untuk melatih *soft skills problem solving*. Aplikasi-aplikasi tersebut menyediakan latihan soal yang harus diselesaikan dengan metode *problem solving*.

Kesulitan dan hambatan dalam melakukan metode berpikir seperti ini tentunya dialami oleh programer. Ketika melakukan proses membuat program, programer dapat mengalami *bug*, yaitu suatu keadaan ketika aplikasi atau program yang dijalankan tidak dapat berfungsi sebagaimana mestinya. Pada saat seperti inilah, programer harus mengulas kembali metode yang dilakukan dari awal. Programer mencari cara untuk mengetahui bagian yang terlewat, menulis masalah yang terjadi, bertukar pikiran dengan rekan atau tim, mencari referensi di internet, membuat diagram, hingga akhirnya dapat mengatasi bagian yang bermasalah.

4. Strategi Membangun Pola Pikir Problem Solving saat Mempelajari Teknologi

Pada era digital yang terus berkembang, kemampuan dalam *problem solving* atau pemecahan masalah menjadi keterampilan yang sangat penting, terutama bagi individu yang belajar teknologi. Menguasai keterampilan ini tidak hanya akan membantu memahami konsep teknologi dengan lebih baik, tetapi juga sebagai persiapan untuk menghadapi tantangan kompleks di dunia nyata. Berikut enam langkah strategis dalam membangun pola pikir *problem solving* saat mempelajari teknologi.