

APLIKASI PEMBELAJARAN ARITMETIKA MENGGUNAKAN METODE BACKUS NAUR FORM (BNF) BERBASIS ANDROID DI SDN RAMPAL CELAKET MALANG

Prayoga Wahyu Setiawan¹⁾, Dwi Safiroh Utsalina²⁾
Program Studi Teknik Informatika, STMIK Pradnya Paramita Malang
prayoga.wahyu@gmail.com¹⁾, utsalina@gmail.com²⁾

Abstract

The user growth of tablets and smartphones among students in Indonesia can be used as an opportunity to help increase interest in learning and understanding a lesson for students. From the results of observations made in SDN Rampil Celaket Malang, Arithmetic subchapter of Mathematics is one of the difficult subject matter to be understood by students and elementary school grades 1 to 3. that's why proposed to construct an learning arithmetic applications based on Android.

Keywords: Application , learning , android .

Abstrak

Pertumbuhan pengguna tablet dan smartphone di kalangan pelajar di Indonesia dapat dimanfaatkan sebagai peluang untuk membantu meningkatkan minat belajar dan pemahaman pelajaran bagi siswa. Dari hasil observasi yang dilakukan di SDN Rampil Celaket Malang, subbab Aritmatika Matematika merupakan salah satu mata pelajaran yang sulit dipahami oleh siswa sekolah dasar kelas 1 sampai 3. Oleh karena itu, diusulkan untuk membangun aplikasi pembelajaran aritmatika berbasis Android.

Kata kunci: Aplikasi, pembelajaran, android.

PENDAHULUAN

Perangkat elektronik berbasis komputer di dunia semakin berkembang pesat, seperti tablet dan smartphone. Tersedianya berbagai macam aplikasi seperti navigasi petunjuk jalan, pemutar musik serta video dan aplikasi yang dapat digunakan untuk media pembelajaran dapat memudahkan kegiatan sehari-hari di kalangan masyarakat, dari dewasa hingga anak-anak. Pengguna tablet dan smartphone berbasis Android di Indonesia semakin meningkat setiap bulannya seperti pada gambar 1

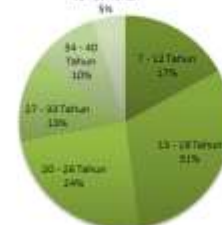


Gambar 1. Data Statistik Pengguna Android di Indonesia

(Sumber: <http://gs.statcounter.com/>)

Berdasarkan hasil survey yang dilakukan oleh Tech in Asia Indonesia pada gambar 1.2 dapat dilihat bahwa pengguna android dikalangan anak SD usia 7-12 tahun mencapai 17% atau hampir 1/5 dari keseluruhan pengguna tablet dan smartphone berbasis android di Indonesia. Meluasnya penggunaan tablet dan smartphone pada kalangan pelajar SD di Indonesia dapat dijadikan peluang untuk membantu meningkatkan minat belajar serta pemahaman akan pelajaran bagi siswa.

Rentang Usia Pengguna Android di Indonesia Tech in Asia Indonesia 2014



Gambar 2. Data Statistik Rentang Usia Pengguna Android di Indonesia

(Sumber: <http://id.techasia.com/>)

Berdasarkan hasil survei di SDN Rampal Celaket Malang dapat dilihat di tabel 1.1, sebanyak 46,9% siswa siswi dari tingkat kelas 1 hingga 6 SD menyatakan bahwa mata pelajaran matematika sukar untuk cepat dipahami, prosentase tertinggi di nyatakan oleh siswa siswi tingkat kelas 1 hingga 3 SD. Subab matematika yang sama dari ketiga kelas tersebut adalah Aritmetika dasar. Aritmetika merupakan cabang matematika yang mempelajari operasi dasar bilangan. Operasi dasar aritmetika adalah penjumlahan, pengurangan, perkalian dan pembagian. Perhitungan dalam aritmetika dilakukan menurut suatu urutan operasi yang menentukan operasi aritmetika yang mana lebih dulu dilakukan.

Backus Naur Form (BNF) adalah *metalanguage* sintaktis yang umum digunakan sebagai notasi untuk mempresentasikan generasi bahasa. Sebagian besar *compiler* seperti Turbo, minGW dan GNU C/C++ menggunakan BNF untuk *parsing script* pemrograman yang ditulis sebelum di *compile* dan dapat dijalankan oleh komputer.

Karena itu, BNF tepat digunakan untuk pemrosesan soal aritmetika menjadi urutan pengerjaan soal yang benar. Dengan menggunakan BNF algoritma yang digunakan tidak terlalu panjang dan rumit serta menghasilkan *output* yang tepat dan tanpa *bug*. Berdasarkan masalah yang telah dipaparkan maka dibuatlah sebuah “Aplikasi Pembelajaran Aritmetika Menggunakan Metode Backus Naur Form (BNF) Berbasis Android Di SDN Rampal Celaket Malang“. Adanya aplikasi ini dapat membantu siswa siswi dalam belajar aritmetika dengan mudah dan menyenangkan.

Berdasarkan latar belakang yang telah dipaparkan, maka dapat dirumuskan permasalahan sebagai berikut:

Bagaimana membangun Aplikasi Pembelajaran Aritmetika Menggunakan Metode

Backus Naur Form (BNF) Berbasis *Android* Di SDN Rampal Celaket Malang?

Dalam penelitian ini membahas beberapa permasalahan sebagai berikut :

1. Aplikasi pembelajaran Aritmetika untuk anak SDN Rampal Celaket kelas 1 hingga 3.
2. Metode yang digunakan Backus Naur Form (BNF).
3. Aplikasi yang digunakan berbasis *Android*.

Adapun tujuan penelitian yang dilakukan adalah :

Terbangunnya Aplikasi Pembelajaran Aritmetika Menggunakan Metode Backus Naur Form (BNF) Berbasis *Android* Di SDN Rampal Celaket Malang. Adapun manfaat yang dapat diperoleh dari penelitian ini adalah sebagai berikut:

1. Bagi SDN Rampal Celaket

Mempermudah siswa siswi tingkat kelas 1 hingga 3 Sekolah Dasar dalam belajar aritmetika, sehingga dapat meningkatkan kemampuan mereka pada mata pelajaran matematika.

2. Bagi Guru atau Pengajar

Sebagai media baru untuk membantu pengajaran aritmetika pada tingkat kelas 1 hingga 3 Sekolah Dasar.

3. Bagi Peneliti

Dapat memberikan pemahaman baru tentang *metalanguage* Backus Naur Form (BNF) sebagai notasi untuk mempresentasikan *grammar* dari bahasa pemrograman atau aturan yang memiliki *pattern* yang lain.

Java

Menurut Suarga (2007:2) Java adalah nama salah satu bahasa pemrograman komputer yang berorientasi objek, diciptakan oleh satu tim dari perusahaan Sun Microsystem, perusahaan workstation UNIX (Sparc) yang cukup terkenal. JAVA diciptakan berdasarkan C++, dengan tujuan platform independent (dapat dijalankan pada berbagai jenis *hardware* tanpa kompilasi ulang),

dengan slogan Write once Run Anywhere (WORA). Dibanding bahasa C++, JAVA pada hakikatnya lebih sederhana dan memakai objek secara murni.

Android

Menurut Safaat, (2012:1) *Android* adalah sebuah sistem operasi pada *handphone* yang bersifat terbuka dan berbasis pada sistem operasi *Linux*. *Android* dapat digunakan oleh setiap orang yang ingin menggunakannya pada perangkat mereka, karena bersifat *Open Source* aplikasi *android* juga dapat dikembangkan secara bebaas dan gratis. *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat selular di lain pihak, *Google* merilis kode-kode *android* di bawah lisensi *apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat selular.

UML (Unified Modelling Language)

UML (*Unified Modelling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. Rosa dan Salahuddin (2013:133).

Menurut Munawar (2005:17), UML (*Unified Modelling Language*) adalah salah satu alat bantu dalam pengembangan sistem berorientasi obyek. UML telah dijadikan sebagai suatu standart baku dalam OOAD (*Object Oriented Analysis and Design*) berdasarkan hasil kerja dari konsorsium berbagai organisasi.

Terdapat tiga fungsi UML, yaitu: Munawar (2005:19)

1. Sebagai sebuah sketsa, yaitu sebagai jembatan dalam mengkomunikasikan berbagai aspek dari sistem.
2. Sebagai sebuah cetak biru, yaitu dapat diketahui informasi detail tentang *coding* program atau

membaca program dan menginterpretasikannya kembali ke dalam diagram.

3. Sebagai bahasa pemrograman, yaitu dapat menerjemahkan *code* program yang siap untuk dijalankan.

Tipe Diagram UML

Menurut Rosa (2013:140-167), terdapat 13 tipe diagram UML. Namun yang akan digunakan dalam penelitian ini hanya empat tipe diagram yang digunakan, yaitu:

Use Case Diagram

Diagram *use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat difahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor atau *use case*.

- Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat sendiri, jadi walaupun simbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang.
- *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol yang ada pada diagram *use case* Rosa (2013:156-158):

Activity Diagram

Diagram *activity* adalah *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang

dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal seperti berikut:

- Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- Urutan atau pengelompokan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antar muka tampilan.
- Rancang pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
- Rancang menu yang ditampilkan pada perangkat lunak.

Class Diagram

Diagram *Class* atau *Class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas.
- Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Sequence Diagram

Diagram *Sequence* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* menolong *analyst* dan programmer untuk memecahkan masalah kedalam

segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

Flowchart adalah bentuk gambar/diagram yang mempunyai aliran satu atau dua arah secara sekuensial. *Flowchart* juga digunakan untuk memecah dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem.

Flowchart yang digunakan dalam menggambarkan sistem dan aplikasi pada penelitian ini adalah jenis *Flowchart* Program. *Flowchart* Program dapat menggambarkan keseluruhan dari urutan atau prosedur sistem yang berjalan sekarang. *Flowchart* tersebut juga dapat mempresentasikan langkah-langkah serta kinerja aplikasi yang dibuat dalam membantu sistem yang sudah berjalan.

Pedoman-Pedoman Dalam Membuat Flowchart

Bila seorang analis dan programmer akan membuat flowchart, ada beberapa petunjuk yang harus diperhatikan, seperti :

1. *Flowchart* digambarkan dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktivitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktivitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari aktivitas harus diuraikan dengan menggunakan deskripsi kata kerja, misalkan menghitung pajak penjualan.
5. Setiap langkah dari aktivitas harus berada pada urutan yang benar.
6. Lingkup dan range dari aktivitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada flowchart yang sama. Simbol konektor harus digunakan dan

percabangannya diletakan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.

7. Gunakan simbol-simbol flowchart.

Materi Aritmetika SD kelas 1 hingga 3

Fajariah (2008:18) Peta konsep materi matematika untuk kelas 3 SD/MI adalah operasi bilangan bulat, operasi penjumlahan dan pengurangan tanpa menyimpan dan dengan menyimpan, nilai tempat sampai ribuan, operasi perkalian dan pembagian dengan nilai tempat sampai ratusan.

Metalanguage Backus Naur Form (BNF)

Menurut Kandaga (2005:61) sebuah bahasa pemrograman didefinisikan dengan cara mensespesifikasikan sintaks (struktur), dan semantic (arti) dari bahasa tersebut. Spesifikasi sebuah bahasa pemrograman biasanya dinyatakan dalam bentuk Backus Naur Form (BNF), *extended* (BNF) atau diagram sintaks.

BNF (Backus Naur Form) adalah *metalanguage* sintaktis yang umum digunakan sebagai notasi untuk mempresentasikan generasi bahasa. Disini terjadi sedikit perbedaan dengan *metalanguage* yang kita gunakan sebelumnya. Disini, *Metavariabel* atau kelas sintaks ditulis dengan diapit simbol < dan >. Jika kita menggunakan notasi ini, maka simbol adalah anggota dari element V_n sedangkan simbol "sentences" adalah element dari V_t . Dengan cara ini kita tidak ragu-ragu melihat apakah suatu simbol merupakan anggota V_n atautkah anggota V_t .

BNF secara intensif telah banyak digunakan dalam pendefinisian secara formal berbagai bahasa pemrograman. Bahasa pemrograman populer yang menggunakan deskripsi BNF adalah ALGOL. Sebagai contoh, pendefinisian sebuah identifier di dalam BNF adalah sebagai berikut :

$:: = | \text{identifier} > | < \text{identifier} <$

$:: = a|b|c|.....|y|z$

$:: = 0|1|2|.....|8|9$

METODE

Metode yang digunakan dalam penelitian ini di antaranya yaitu:

Identifikasi masalah

Melakukan identifikasi masalah pada sistem dan cara belajar mengajar yang sedang berjalan pada saat ini.

Pengumpulan Data

Pengumpulan dan analisis data dilakukan dalam penyusunan Tugas Pemrograman Khusus ini terdiri dari:

i. Wawancara

Pengumpulan data dilakukan dengan mengajukan beberapa pertanyaan pada Wali Kelas terkait nilai siswa siswi di setiap mata pelajaran dan kendala yang di alami dalam proses belajar mengajar.

ii. Observasi

Penelitian ini dimaksudkan untuk mengetahui lebih jelas kemampuan serta motivasi belajar siswa siswi di SDN Rampal Celaket Malang.

iii. Studi literatur

Pengumpulan data dilakukan dengan mempelajari teori-teori dari buku-buku dan jurnal dari hasil penelitian sebelumnya yang mendukung materi penelitian sebagai sumber acuan dan pendalaman landasan teori dalam pengembangan aplikasi.

Analisis

Tahapan ini dilakukan untuk mengetahui masalah yang dihadapi dalam proses belajar mengajar aritmetika yang sekarang sedang berjalan dan membandingkan perubahan proses tersebut sebelum dan sesudah menggunakan aplikasi pembelajaran dalam prosesnya.

Desain dan Rancangan Aplikasi

Setelah data terkumpul dan dianalisis, tahap berikutnya melakukan pendesainan dan perancangan aplikasi yang meliputi perancangan form program, *flow chart*, perancangan skenario *Use Case*, *Activity Diagram* dan *Class Diagram*.

Evaluasi Program

Evaluasi program dilakukan untuk mengetahui adanya kekurangan atau kelemahan aplikasi, sehingga dapat dilakukan perbaikan.

Rancang Bangun Program Pada tahap ini dilakukan proses pembangunan desain (*interface* tampilan sistem), penyusunan *script* program.

Pengujian Aplikasi

Pengujian pada aplikasi ini dilakukan dengan cara menguji perhitungan matematis untuk mengetahui urutan serta jawaban yang benar. Pengujian kembali akan dilakukan jika ditemukan kesalahan dalam penyusunan *code* program yang sedang berjalan.

Alat dan Bahan Penelitian

Penelitian ini menggunakan alat penelitian berupa perangkat keras dan perangkat lunak, yaitu

1. Perangkat Keras

- Processor Intel core i5-4200 1.60Hz (4 CPU)
- Besar memory RAM 4096MB
- Kapasitas Harddisk 500GB
- Monitor dengan resolusi 1366 x 768 px.
- Perangkat mouse dan keyboard standar
- Ponsel Samsung Galaxy S4 i-9500

2. Perangkat Lunak

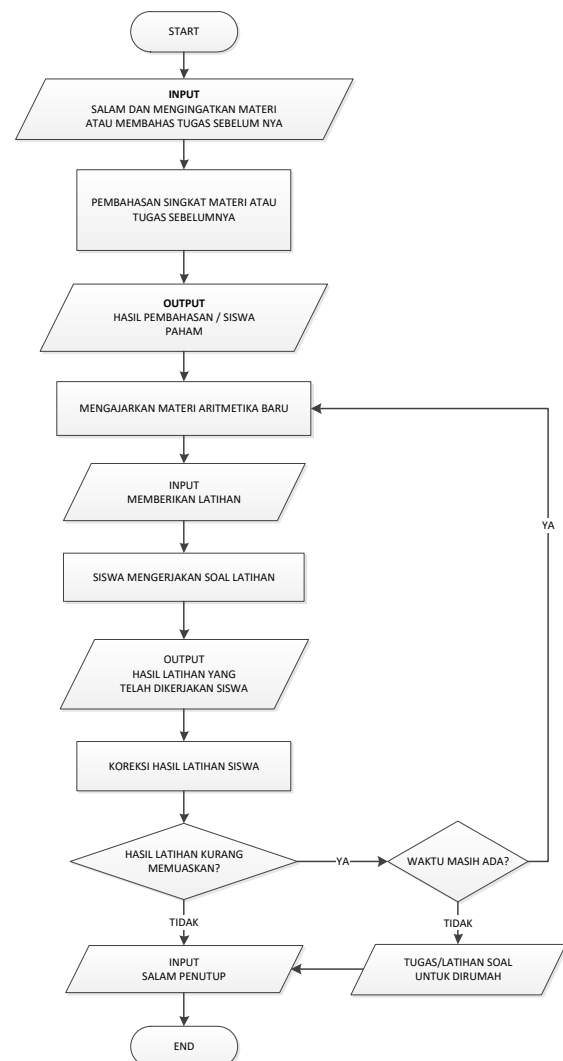
- Sistem Operasi Microsoft Windows 7 32bit
- Sistem operasi Android 4.3 Jelly Bean
- Java development Kit (JDK) jdk1.8.0_20
- Android Development Tool
- Android Standart Development Kit (SDK)

HASIL DAN PEMBAHASAN

SDN Rampal Celaket Malang adalah Sekolah Dasar Negeri yang terdapat di tengah Kota Malang. Terletak pada jalan WR Supratman no 11

A malang, Sekolah ini memiliki 6 kelas dan 226 siswa aktif pada tahun 2014.

Berdasarkan hasil observasi di SDN Rampal Celaket Malang dapat dilihat di tabel 1.1, sebanyak 46,9% siswa siswi dari tingkat kelas 1 hingga 6 SD menyatakan bahwa mata pelajaran matematika sukar untuk cepat dipahami, prosentase tertinggi di nyatakan oleh siswa siswi tingkat kelas 1 hingga 3 SD. Subab matematika yang sama dari ketiga kelas tersebut adalah Aritmetika dasar.



Gambar 4. *Flow Chart* Proses Pembelajaran Aritmetika di SDN Rampal Celaket

Flowchart Proses Pembelajaran Aritmetika di SDN Rampal Celaket pada gambar 3.1 dapat dideskripsikan sebagai berikut :

1. Tahap pembukaan, awal pertemuan dimulai dengan berdoa dan mengucapkan salam,

2. Kemudian guru akan mengulas secara singkat materi yang sebelumnya diajarkan kepada para siswa. Jika pada pertemuan sebelumnya guru memberikan tugas, pada tahap inilah pembahasan tugas dilakukan. Hasil dari pembahasan materi sebelumnya atau tugas, agar siswa paham

3. mengenai materi yang telah disampaikan sebelum guru memberikan materi selanjutnya.

4. Guru mengajarkan materi baru kepada siswa

5. Setelah materi selesai disampaikan, guru memberikan latihan soal tentang materi yang baru saja diajarkan kepada siswa.

6. Siswa mengerjakan latihan soal dengan didampingi oleh guru.

7. Siswa selesai mengerjakan soal latihan dalam jangka waktu yang telah ditentukan.

8. Guru memeriksa hasil latihan yang telah dikerjakan oleh siswa. Hal ini bertujuan sebagai parameter tingkat pemahaman siswa

9. Jika hasil menunjukkan bahwa siswa masih belum paham dan masih ada waktu pelajaran

10. Maka guru akan kembali mengulang materi yang telah diajarkan dan memberikan latihan kecil contoh soal hingga siswa paham

11. Jika waktu sudah habis maka guru akan memberikan latihan soal tentang materi yang baru saja diajarkan untuk dikerjakan di rumah. Dan kegiatan pembelajaran selesai.

12. Jika hasil menunjukkan bahwa siswa sudah paham akan materi yang baru saja

diajarkan maka kegiatan pembelajaran selesai.

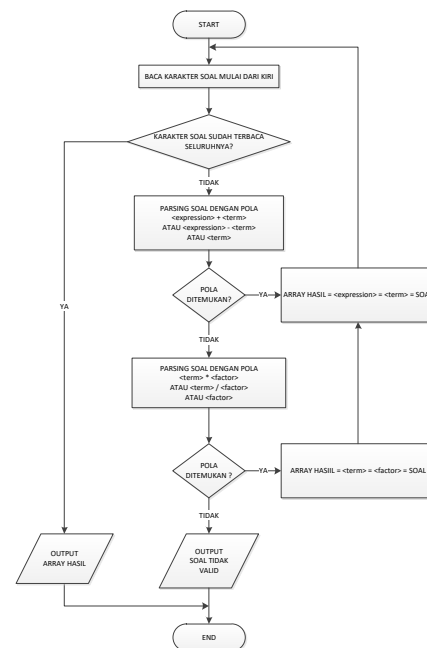
Keterbatasan waktu dan perbedaan kemampuan pemahaman materi diantara siswa, membuat guru kesulitan dalam membimbing siswa nya agar memiliki tingkat pemahaman materi yang sama sebelum memulai mengajar materi selanjutnya. Selalu diulangnya materi yang disampaikan jika siswa masih belum paham juga menghabiskan banyak waktu.

Aplikasi Pembelajaran Aritmetika

Aplikasi pembelajaran aritmetika menggunakan metode *Bakus Naur Form* berbasis *android* untuk kelas 1 hingga 3 Sekolah dasar Rampal Celaket Malang. Aplikasi ini merupakan aplikasi yang dapat membantu proses pengajaran aritmetika kepada siswa kelas 1 hingga 3 SD. Materi aritmetika yang diajarkan pada siswa tingkat kelas 1 hingga 3 Sekolah Dasar adalah operasi matematika penambahan, pengurangan, perkalian dan pembagian dengan maksimal tiga digit angka.

Konsep Implementasi *Bakus Naur Form* (BNF)

Konsep Implementasi *Bakus Naur Form* dapat digambarkan pada alur *Flow Chart* di bawah ini :



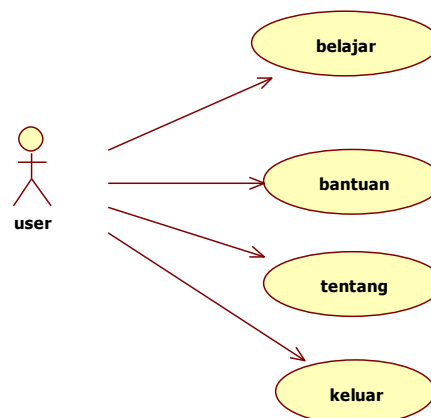
Gambar 5. *Flow Chart* algoritma parsing soal aritmetika dengan menggunakan metode BNF

Flow Chart algoritme *parsing* soal aritmetika dalam aplikasi Pembelajaran Aritmetika di SDN Rampal Celaket dapat dideskripsikan sebagai berikut :

1. Memindai karakter soal dimulai dari bagian paling kiri
2. Memeriksa apakah ruas soal sudah habis diolah
3. Jika masih ada ruas yang belum diolah maka proses *parsing* soal aritmetika dengan pola $\langle \text{expression} \rangle + \langle \text{term} \rangle$ atau $\langle \text{expression} \rangle - \langle \text{term} \rangle$ atau $\langle \text{term} \rangle$ dijalankan
4. Jika pola pada proses poin 3 ditemukan maka hasil sementara disimpan di array hasil, kemudian proses dilanjutkan kembali ke poin 2.
5. Jika pola tidak ditemukan maka proses berlanjut ke *parsing* soal aritmetika dengan pola $\langle \text{term} \rangle * \langle \text{factor} \rangle$ atau $\langle \text{term} \rangle / \langle \text{factor} \rangle$ atau $\langle \text{factor} \rangle$
6. Jika pola pada proses poin 6 ditemukan maka hasil sementara disimpan di array hasil, kemudian proses dilanjutkan kembali ke poin 2.
7. Jika kedua pola tersebut tidak ditemukan maka soal tidak valid, aplikasi menampilkan peringatan soal tidak valid.
8. Jika ruas soal seluruhnya sudah selesai diproses maka aplikasi menampilkan hasil proses yang terdapat pada array hasil

Use Case Diagram

Pemodelan *use case* diagram pada Aplikasi Pembelajaran Aritmetika Menggunakan Metode Backus Naur Form (BNF) Berbasis Android Di SDN Rampal Celaket Malang diilustrasikan pada gambar berikut ini:



Gambar 6. *Use Case* Diagram

Use case diagram terdiri atas beberapa obyek yaitu *actor* dan *use case*. *Actor* dan *use case* pada Aplikasi Pembelajaran Aritmetika Menggunakan Metode Backus Naur Form (BNF) Berbasis Android Di SDN Rampal Celaket Malang dijabarkan pada identifikasi *actor* dan identifikasi *use case* berikut ini:

A. Identifikasi *Actor*

Actor memperlihatkan himpunan pengguna *use case*, himpunan ‘sesuatu’ yang berinteraksi dengan sistem yang dikembangkan Nugroho (2005:90). Dalam aplikasi ini *actor* pengguna aplikasi adalah User.

User adalah pengguna umum aplikasi, user ini dapat menggunakan aplikasi untuk belajar urutan mengerjakan soal aritmetika, melihat bantuan cara menggunakan aplikasi, melihat informasi tentang aplikasi dan keluar dari aplikasi.

B. Identifikasi *Use Case*

Use case digunakan untuk menspesifikasikan perilaku sistem (atau bagian dari sistem secara keseluruhan) dan merupakan deskripsi dari sekumpulan aksi-aksi yang diharapkan oleh calon pengguna sistem/perangkat lunak yang dikembangkan (Nugroho, 2005:89).

Dalam aplikasi ini *use case* yang diidentifikasi meliputi:

1. Belajar

Proses yang dilakukan oleh *user* untuk belajar mengerjakan soal aritmetika sesuai dengan urutan yang tepat.

2. Bantuan

Proses untuk menampilkan bantuan kepada *user* tentang cara menggunakan aplikasi pembelajaran aritmetika ini.

3. Tentang

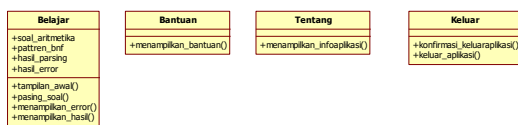
Proses untuk menampilkan informasi kepada *user* tentang aplikasi pembelajaran aritmetika ini.

4. Keluar

Proses yang dapat dilakukan oleh *user* untuk keluar atau menutup aplikasi pembelajaran aritmetika ini.

Class Diagram

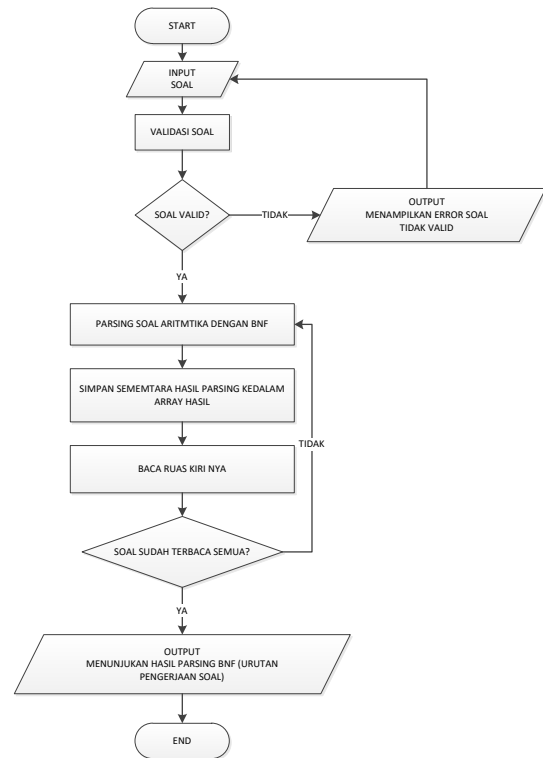
Class diagram sangat membantu dalam visualisasi struktur kelas dari sistem dan memberikan pandangan global atas sebuah sistem. *Class* adalah deskripsi kelompok obyek-obyek dengan *property*, perilaku (operasi) dan relasi yang sama Munawar (2005:219). Pada Gambar 6 berikut adalah *class* diagram dari aplikasi pembelajaran aritmetika menggunakan *Backus Naur Form* berbasis *android* di SDN Rampil Celaket Malang



Gambar 6. *Class* Diagram

Flow Chart Program

Alur dari proses *input output* hingga *parsing* soal aritmetika menggunakan *Backus Naur Form* dalam aplikasi yang akan dibangun ini digambarkan dalam bentuk Diagram Alir (*Flow Chart*) sebagai berikut :



Gambar 7. *Flow Chart* Proses *input output* soal aritmetika dalam aplikasi

Flow Chart Proses *input output* soal aritmetika dalam aplikasi Pembelajaran Aritmetika di SDN Rampil Celaket pada gambar 3.8 dapat dideskripsikan sebagai berikut :

1. *User* memasukkan soal aritmetika ke dalam kolom soal yang disediakan
2. Proses validasi soal aritmetika
3. Jika soal tidak valid maka aplikasi akan menampilkan pesan bahwa soal tidak valid dan *user* disarankan untuk memeriksa soal kembali
4. Jika soal valid maka proses *parsing* soal menggunakan BNF dijalankan
5. Hasil proses *parsing* sementara disimpan kedalam *variable array* hasil
6. Ruas soal yang sudah di olah di hilangkan
7. Memeriksa apakah ruas soal sudah habis diproses

8. Jika belum kembali ke algoritma pemrosesan *parsing* menggunakan BNF hingga soal aritmetika seluruhnya selesai diproses.

9. Menampilkan hasil *parsing* yang telah disimpan dalam *array* hasil

Hasil Pengujian Aplikasi

A. Menu Belajar

1. Rencana Pengujian Menu Belajar

Tabel 1 Rencana Pengujian Menu Belajar

No.	Input	Output	Kesesuaian
1.	Soal : 123	Terjadi Kesalahan!, soal tidak benar.	Ya
2.	Soal : 40*4:	Terjadi Kesalahan!, soal tidak benar.	Ya
3.	Soal : +4*6	Terjadi Kesalahan!, soal tidak benar.	Ya
4.	Soal : 6+4	1). = (6+4) 2). = 10	Ya
5.	Soal : 5- 3+7	1). = (5-3)+7 2). = (2+7) 3). = 9	Ya
6.	Soal : 1+2:*6	Terjadi Kesalahan!, soal tidak benar.	Ya
7.	Soal : 4*5+6	1). = (4*5)+6 2). = (20+6) 3). = 26	Ya
8.	Soal : 1000++++ 5	Terjadi Kesalahan!, soal tidak benar.	Ya
9.	Soal : 1*3- 2:2	1). = (1*3)-2:2 2). = 3-(2:2) 3). = (3-1) 4). = 2	Ya
10.	Soal : 579*2-90	1). = (579*2)-90 2). = (1158-90) 3). = 1068	Ya

2. Hasil Pengujian Menu belajar

Tampilan Awal Menu Belajar

Pada tampilan menu belajar, *user* disajikan form yang memiliki kotak input untuk memasukkan soal yang akan dihitung.

KESIMPULAN DAN SARAN

Berdasarkan pembahasan pada bab sebelumnya maka, dapat disimpulkan hasil observasi yang dilakukan di SDN Rampal Celaket Malang, Aritmetika subab dari Matematika adalah salah satu materi pelajaran yang sulit untuk dipahami oleh siswa dan siswi SD kelas 1 hingga 3. Terdapat berbagai kendala yang dialami oleh guru dalam proses belajar mengajar aritmetika. Keterbatasan waktu dan perbedaan kemampuan pemahaman materi diantara siswa, membuat guru kesulitan dalam membimbing siswa nya agar memiliki tingkat pemahaman materi yang sama sebelum memulai mengajar materi selanjutnya. Aplikasi pembelajaran aritmetika menggunakan metode *Bakus Naur Form* berbasis *android* yang diberi nama “HITUNG ARITMETIKA” atau disingkat “HIKA” ini merupakan aplikasi yang dapat membantu proses pengajaran aritmetika kepada siswa kelas 1 hingga 3 SD. Terbukti dengan setelah dilakukannya ujicoba penerapan aplikasi ini, rata-rata siswa yang bertanya dan rata-rata pengulangan materi yang telah disampaikan oleh guru menurun sekitar 50%. Saran untuk penelitian berikutnya dengan memperhatikan pembahasan, kesimpulan dan keterbatasan penelitian adalah penyesuaian *user interface* yang lebih menarik serta lebih *user friendly* kepada anak – anak tingkat SD. Menambahkan operasi aritmetika perpangkatan, pengakaran dan operasi prioritas tanda kurung, meningkatkan cakupan perhitungan hingga satuan desimal dan minus agar aplikasi yang dibangun dapat digunakan oleh oleh tingkat kelas 1 hingga 6 SD . Menyangkut objek penelitian juga dapat dikembangkan kepada objek yang lebih umum

sehingga manfaat dari aplikasi yang dibangun juga dapat dirasakan oleh masyarakat luas.

REFRENSI

1. Dhanta, Rizky. 2009. *Pengantar Ilmu Komputer*. Surabaya: Indah.
2. Djamarah, Syaiful Bahri dan Aswan Zain. 2010. *Strategi Belajar Mengajar*. Jakarta: Rineka Cipta.
3. Fajariyah, Nur 2008. *Cerdas Berhitung Matematika Untuk SD/MI Kelas 3*. Jakarta: Pusat Pembukuan Departemen Pendidikan Nasional.
4. Harahap, ST dan B Negoro. 2010. *Ensiklopedi Matematika*. Jakarta: Galia Indonesia IKAPI.
5. Inmon, William H. 2005. *Building The Data Warehouse*. New York : Wiley Computer Publishing
6. Juarna, Asep, 2006. *Catatan Teori Bahasa dan Atutomata*. Jakarta: Universitas Gunadarma
7. Kandaga, Tjatur. 2005. *Analisis dan Perancangan Bahasa Pemrograman Paralel Beserta Pembuatan Prototipe Kompilatornya*. Jurnal Informatika UKM, 1 (1): 53-76
8. Mudjiono dan Dimjati. 2009. *Belajar dan Pembelajaran*. Jakarta: Rineka Cipta.
9. Safaat, H Nazaruddin, *Sistem Informasi Georafis Berbasis Mobile*, Yogyakarta: Gava Media.
10. Suarga. 2009. *Dasar Pemrograman Komputer Dalam Bahasa Java*. Yogyakarta : Andi Offset.
11. S, Rosa dan Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak*. Bandung: Informatika.
12. Sulindawati dan Fathoni Muhammad. 2010. *Pengantar Analisa Perancangan Sistem*, 9 (2): 8–18.
13. Slameto. 2010. *Belajar dan Faktor – Faktor Yang Mempengaruhinya*. Jakarta: Bina ksara.
14. Sudjana, Nana. 2010. *Dasar-Dasar Proses Belajar Mengajar*. Bandung: Sinar Baru.
15. Zulfa, Umi. 2010. *Strategi Pembelajaran (Edisi Revisi)*. Cilacap: Al Ghazali Press