**WEB SMK NEGERI 4 BANDUNG  
BERBASIS PHP dan MYSQL**

Tugas Akhir Pemrograman WEB



Disusun Oleh :

0910113314

Egi Soleh Hasdi

PROGRAM KEAHLIAN TEKNIK KOMPUTER DAN INFORMATIKA  
KOMPETENSI KEAHLIAN REKAYASA PERANGKAT LUNAK  
SMK NEGERI 4 BANDUNG  
2012

**LEMBAR PENGESAHAN**

**WEB SMK NEGERI 4 BANDUNG**

**Disetujui di Bandung, Maret 2012**

**Pembimbing I Pembimbing II**

**Kurnia Jaya Eliazar, ST Lukmanul Hakim F.**

**NIP. NIP.**

# ABSTRAK

Dengan semakin pesatnya kemajuan teknologi informasi dan komunikasi maka semakin mudahnya orang orang untuk mengakses informasi melalui media yang terhubung dengan internet. Salah satu dari media informasi yaitu dengan adanya website yang dapat diakses oleh semua orang melalui berbagai macam perangkat mulai dari handphone sampai komputer. Ini merupakan kesempatan bagi suatu instansi/sekolah untuk dapat mempublikasikan informasi informasi yang dibutuhkan siswa/masyarakat. Dan juga merupakan media promosi agar masyarakat dapat lebih mengenal sekolah tersebut.

Kata kunci: Teknologi Informasi dan komunikasi, Website, Internet.

# KATA PENGANTAR

Assalamu’alaikum Wr. Wb.

Puji dan Syukur penulis ucapkan dan panjatkan ke hadirat Tuhan Yang Maha Esa, yang selalu memberikan kesempatan, kesehatan, dan kelancaran sehingga pada akhirnya **Tugas Akhir** ini dapat diselesaikan dalam waktu yang telah di tetapkan.

Penulisan **Tugas Akhir** ini merupakan salah satu syarat utama kelulusan dalam menempuh pendidikan di jurusan Rekayasa Perangkat Lunak. Pada kesempatan ini Penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Orang Tua yaitu Bapak dan Ibu yang selalu mendukung dan berdo’a demi kesuksesan penulis.
2. Drs. Asep Dada Wahyudin, M.Mpd selaku Kepala Sekolah SMKN 4 Bandung.
3. Bapak Dadan Hermawan selaku Ketua Jurusan Informatika RPL/TKJ SMKN 4 Bandung.
4. Bapak Kurnia dan Ibu Hana selaku pengajar dan pembimbing yang selalu sabar dalam membimbing dan menuntun pembuatan **Tugas Akhir Java**ini.
5. Berbagai pihak yang tidak dapat penulis sebutkan satu-persatu, Terima Kasih untuk semua bantuan dan dukungannya.

Akhir kata penulis berharap semoga **Tugas Akhir Java** ini dapat berguna bagi semua pihak yang membaca. Terima kasih.

Wassalamu’alaikum Wr. Wb.s

Bandung, Maret 2012

Penulis

# DAFTAR ISI

[ABSTRAK i](#_Toc320472277)

[KATA PENGANTAR ii](#_Toc320472278)

[DAFTAR ISI iii](#_Toc320472279)

[DAFTAR GAMBAR vi](#_Toc320472280)

[DAFTAR TABEL vii](#_Toc320472281)

[DAFTAR LAMPIRAN viii](#_Toc320472282)

[BAB I PENDAHULUAN 1](#_Toc320472283)

[1.1 Latar Belakang 1](#_Toc320472284)

[1.2 Rumusan Masalah 1](#_Toc320472285)

[1.3 Batasan Masalah 2](#_Toc320472286)

[1.4 Sistematika Penulisan 2](#_Toc320472287)

[BAB II TEORI DASAR 3](#_Toc320472288)

[2.1 Game 3](#_Toc320472289)

[2.1.1 Syarat Game 3](#_Toc320472290)

[2.1.2 Jenis-Jenis Game 6](#_Toc320472291)

[2.2 Java 9](#_Toc320472292)

[2.2.1 Sejarah singkat Java 9](#_Toc320472293)

[2.2.2 Karakteristik Java 10](#_Toc320472294)

[2.2.3 Java Virtual Machine (JVM) 12](#_Toc320472295)

[2.2.4 Garbage Collection 13](#_Toc320472296)

[2.2.5 Code Security 13](#_Toc320472297)

[2.3 J2ME (Java 2 Micro Edition) 14](#_Toc320472298)

[2.3.1 Configuration 15](#_Toc320472299)

[2.3.2 Profile 16](#_Toc320472300)

[2.3.3 CLDC 16](#_Toc320472301)

[2.3.4 Fitur Yang Hilang 16](#_Toc320472302)

[2.3.5 Karakteristik Perangkat CLDC 17](#_Toc320472303)

[2.3.6 Verifikasi Class 17](#_Toc320472304)

[2.3.7 Generic Connection Framework 18](#_Toc320472305)

[2.3.8 CDC 19](#_Toc320472306)

[2.3.9 JTWI 20](#_Toc320472307)

[2.3.10 MIDP 20](#_Toc320472308)

[2.3.11 MIDLET 21](#_Toc320472309)

[2.3.12 Siklus MIDlet 22](#_Toc320472310)

[2.3.13 MIDlet Suites 23](#_Toc320472311)

[2.4 Netbeans IDE 23](#_Toc320472312)

[BAB III ANALISIS MASALAH 24](#_Toc320472313)

[3.1 Use Case 24](#_Toc320472314)

[3.2 Activity Diagram 24](#_Toc320472315)

[3.3 Analisis Kebutuhan Sistem 25](#_Toc320472316)

[3.3.1 Spesifikasi Software 25](#_Toc320472317)

[3.3.2 Spesifikasi Hardware 26](#_Toc320472318)

[3.3.3 Spesifikasi Handphone 26](#_Toc320472319)

[BAB IV PERANCANGAN 27](#_Toc320472320)

[4.1 Perancangan Tampilan 27](#_Toc320472321)

[4.1.1 Tampilan *Splash Screen* 27](#_Toc320472322)

[4.1.2 Perancangan tampilan menu utama 28](#_Toc320472323)

[4.1.3 Perancangan tampilan petunjuk 28](#_Toc320472324)

[4.1.4 Perancangan tampilan tentang 29](#_Toc320472325)

[4.1.5 Perancangan tampilan “*Game Over”* 30](#_Toc320472326)

[4.2 Gambaran Struktur Tampilan 31](#_Toc320472327)

[4.3 Jadwal Kerja 32](#_Toc320472328)

[4.3.1 Gantt Chart 32](#_Toc320472329)

[BAB V IMPLEMENTASI 33](#_Toc320472330)

[5.1 Implementasi Tampilan 33](#_Toc320472331)

[5.1.1 Tampilan *Splash Screen* 33](#_Toc320472332)

[5.1.2 Tampilan Menu Utama 37](#_Toc320472333)

[5.1.3 Tampilan Petunjuk 41](#_Toc320472334)

[5.1.4 Tampilan Tentang 43](#_Toc320472335)

[5.1.5 Tampilan Game Over 45](#_Toc320472336)

[5.2 Implementasi Game 47](#_Toc320472337)

[BAB VI PENUTUP 48](#_Toc320472338)

[6.1 Kesimpulan 48](#_Toc320472339)

[6.2 Saran 48](#_Toc320472340)

[DAFTAR PUSTAKA 49](#_Toc320472341)

[LAMPIRAN A SOURCE CODE A-1](#_Toc320472342)

[LAMPIRAN B BIODATA PENULIS B-1](#_Toc320472343)

# DAFTAR GAMBAR

[Gambar 2.1 Arsitektur J2ME 15](#_Toc320472210)

[Gambar 2.2 Proses Verifikasi dua tingkat 18](#_Toc320472211)

[Gambar 2.3 Hirarki koneksi GCF 18](#_Toc320472212)

[Gambar 2.4 Komponen JTWI 20](#_Toc320472213)

[Gambar 2.5 Daur Hidup MIDlet 22](#_Toc320472214)

[Gambar 3.1 Pemodelan Use Case 24](#_Toc320472215)

[Gambar 3.2 Pemodelan Activity Diagram 25](#_Toc320472216)

[Gambar 4.1 Perancangan tampilan *splash screen* 27](#_Toc320472217)

[Gambar 4.2 Perancangan tampilan menu utama 28](#_Toc320472218)

[Gambar 4.3 Perancangan tampilan petunjuk 29](#_Toc320472219)

[Gambar 4.4 Perancangan tampilan tentang 30](#_Toc320472220)

[Gambar 4.5 Perancangan tampilan “*Game Over”* 31](#_Toc320472221)

[Gambar 4.6 Perancangan struktur tampilan 31](#_Toc320472222)

[Gambar 5.1 Implementasi tampilan *splash screen* 33](#_Toc320472223)

[Gambar 5.2 Implementasi tampilan menu utama 37](#_Toc320472224)

[Gambar 5.3 Implementasi tampilan petunjuk 41](#_Toc320472225)

[Gambar 5.4 Implementasi tampilan tentang 43](#_Toc320472226)

[Gambar 5.5 Implementasi tampilan game over 45](#_Toc320472227)

[Gambar 5.6 Implementasi game “Si Kancil” 47](#_Toc320472228)

# DAFTAR TABEL

[Tabel 4.1 Gantt Chart 32](#_Toc320471671)

# DAFTAR LAMPIRAN

[LAMPIRAN A SOURCE CODE A-1](#_Toc320471488)

[A.1 MidletSiKancil.java A-1](#_Toc320471489)

[A.2 MainMenu.java A-3](#_Toc320471490)

[A.3 Petunjuk.java A-6](#_Toc320471491)

[A.4 Tentang.java A-7](#_Toc320471492)

[A.5 GameCanvasSiKancil.java A-8](#_Toc320471493)

[A.6 GameOverCanvas.java A-17](#_Toc320471494)

[A.7 SpriteAnimator.java A-18](#_Toc320471495)

[A.8 SpritesRandomMovement.java A-19](#_Toc320471496)

[A.9 TileAnimator.java A-23](#_Toc320471497)

[LAMPIRAN B BIODATA PENULIS B-1](#_Toc320471498)

# PENDAHULUAN

## Latar Belakang

Dengan banyaknya perangkat yang dapat terhubung ke internet maka pengaksesan informasi akan sangat mudah dan cepat. Salah satu penggunaan internet yaitu untuk mengakses sebuah situs untuk mendapatkan informasi. Maka ini merupakan suatu kesempatan untuk sekolah dapat mempublikasikan atau memperkenalkan sekolah tersebut serta sebagai media untuk memberikan informasi yang dengan mudah diakses oleh semua orang.

## Rumusan Masalah

Sekolah pasti membutuhkan media untuk mempublikasikan informasi atau media untuk mempromosikan dan memperkenalkan sekolah tersebut, jadi media apakah yang tepat untuk memenuhi kebutuhan tersebut ?

## Batasan Masalah

1. Fitur yang terdapat dalam web SMKN4 meliputi manajemen:
   1. Berita/Artikel
   2. Galeri gambar dan video
   3. Majalah dinding
   4. Agenda
   5. Pengguna
   6. Highlight
   7. Polling
   8. Pegumuman
   9. Komentar
   10. Halaman ekstrakulikuler(Hanya Angklung)
   11. Virtual Tour
   12. Buku tamu
   13. Kalender akademik
   14. Statistik
   15. Log Admin

## Sistematika Penulisan

Bab 1 berisi latar belakang, rumusan masalah, batasan masalah dan sistematika penulisan.

Bab 2 berisi teori dasar tentang Website, PHP, dan MYSQL,

Bab 3 berisi analisis masalah dengan use case, activity diagram, dan analisis kebutuhan sistem.

Bab 4.

Bab 5 berisi implementasi tampilan, dan implementasi game.

Bab 6 berisi penutup, kesimpulan dan saran.

# TEORI DASAR

## Internet

Internet (kependekan dari interconnection-networking) secara harfiah ialah sistem global dari seluruh jaringan komputer yang saling terhubung menggunakan standar Internet Protocol Suite (TCP/IP) untuk melayani miliaran pengguna di seluruh dunia. Manakala Internet (huruf 'I' besar) ialah sistem komputer umum, yang berhubung secara global dan menggunakan TCP/IP sebagai protokol pertukaran paket (packet switching communication protocol). Rangkaian internet yang terbesar dinamakan Internet. Cara menghubungkan rangkaian dengan kaedah ini dinamakan internetworking.

### Sejarah Internet

Internet merupakan jaringan komputer yang dibentuk oleh Departemen Pertahanan Amerika Serikat di tahun 1969, melalui proyek ARPA yang disebut ARPANET (Advanced Research Project Agency Network), di mana mereka mendemonstrasikan bagaimana dengan hardware dan software komputer yang berbasis UNIX, kita bisa melakukan komunikasi dalam jarak yang tidak terhingga melalui saluran telepon.

Proyek ARPANET merancang bentuk jaringan, kehandalan, seberapa besar informasi dapat dipindahkan, dan akhirnya semua standar yang mereka tentukan menjadi cikal bakal pembangunan protokol baru yang sekarang dikenal sebagai TCP/IP (Transmission Control Protocol/Internet Protocol).

Tujuan awal dibangunnya proyek itu adalah untuk keperluan militer. Pada saat itu Departemen Pertahanan Amerika Serikat (US Department of Defense) membuat sistem jaringan komputer yang tersebar dengan menghubungkan komputer di daerah-daerah vital untuk mengatasi masalah bila terjadi serangan nuklir dan untuk menghindari terjadinya informasi terpusat, yang apabila terjadi perang dapat mudah dihancurkan.

Pada mulanya ARPANET hanya menghubungkan 4 situs saja yaitu Stanford Research Institute, University of California, Santa Barbara, University of Utah, di mana mereka membentuk satu jaringan terpadu di tahun 1969, dan secara umum ARPANET diperkenalkan pada bulan Oktober 1972. Tidak lama kemudian proyek ini berkembang pesat di seluruh daerah, dan semua universitas di negara tersebut ingin bergabung, sehingga membuat ARPANET kesulitan untuk mengaturnya.

Oleh sebab itu ARPANET dipecah manjadi dua, yaitu "MILNET" untuk keperluan militer dan "ARPANET" baru yang lebih kecil untuk keperluan non-militer seperti, universitas-universitas. Gabungan kedua jaringan akhirnya dikenal dengan nama DARPA Internet, yang kemudian disederhanakan menjadi Internet.

### Internet pada saat ini

Internet dijaga oleh perjanjian bilateral atau multilateral dan spesifikasi teknikal (protokol yang menerangkan tentang perpindahan data antara rangkaian). Protokol-protokol ini dibentuk berdasarkan perbincangan Internet Engineering Task Force (IETF), yang terbuka kepada umum. Badan ini mengeluarkan dokumen yang dikenali sebagai RFC (Request for Comments). Sebagian dari RFC dijadikan Standar Internet (Internet Standard), oleh Badan Arsitektur Internet (Internet Architecture Board - IAB). Protokol-protokol Internet yang sering digunakan adalah seperti, IP, TCP, UDP, DNS, PPP, SLIP, ICMP, POP3, IMAP, SMTP, HTTP, HTTPS, SSH, Telnet, FTP, LDAP, dan SSL.

Beberapa layanan populer di Internet yang menggunakan protokol di atas, ialah email/surat elektronik, Usenet, Newsgroup, berbagi berkas (File Sharing), WWW (World Wide Web), Gopher, akses sesi (Session Access), WAIS, finger, IRC, MUD, dan MUSH. Di antara semua ini, email/surat elektronik dan World Wide Web lebih kerap digunakan, dan lebih banyak servis yang dibangun berdasarkannya, seperti milis (Mailing List) dan Weblog. Internet memungkinkan adanya servis terkini (Real-time service), seperti web radio, dan webcast, yang dapat diakses di seluruh dunia. Selain itu melalui Internet dimungkinkan untuk berkomunikasi secara langsung antara dua pengguna atau lebih melalui program pengirim pesan instan seperti Camfrog, Pidgin (Gaim), Trilian, Kopete, Yahoo! Messenger, MSN Messenger Windows Live Messenger, Twitter, dan lain sebagainya.

Beberapa servis Internet populer yang berdasarkan sistem tertutup (Proprietary System), adalah seperti IRC, ICQ, AIM, CDDB, dan Gnutella.

### Tata tertib Internet

Sama seperti halnya sebuah komunitas, Internet juga mempunyai tata tertib tertentu, yang dikenal dengan nama Nettiquette atau dalam bahasa Indonesia dikenal dengan istilah netiket.

Untuk di Indonesia selain tata tertib sosial di Internet juga diberlakukan peraturan (UU ITE).

### Isu moral dan undang-undang

Terdapat kebimbangan masyarakat tentang Internet yang berpuncak pada beberapa bahan kontroversi di dalamnya. Pelanggaran hak cipta, pornografi, pencurian identitas, dan pernyataan kebencian (hate speech), adalah biasa dan sulit dijaga. Hingga tahun 2007, Indonesia masih belum memiliki Cyberlaw, padahal draft akademis RUU Cyberlaw sudah dibahas sejak tahun 2000 oleh Ditjen Postel dan Deperindag. UU yang masih ada kaitannya dengan teknologi informasi dan telekomunikasi adalah UU Telekomunikasi tahun 1999.

Internet juga disalahkan oleh sebagian orang karena dianggap menjadi sebab kematian. Brandon Vedas meninggal dunia akibat pemakaian narkotik yang melampaui batas dengan semangat dari teman-teman chatting IRCnya. Shawn Woolley bunuh diri karena ketagihan dengan permainan online, Everquest. Brandes ditikam bunuh, dan dimakan oleh Armin Meiwes setelah menjawab iklan dalam Internet.

### Akses Internet

Anak-anak sedang menggunakan komputer untuk mengakses Internet.

Negara dengan akses Internet yang terbaik termasuk Korea Selatan (50% daripada penduduknya mempunyai akses jalurlebar - Broadband), dan Swedia. Terdapat dua bentuk akses Internet yang umum, yaitu dial-up, dan jalurlebar. Di Indonesia, seperti negara berkembang dimana akses Internet dan penetrasi PC sudah cukup tinggi dengan didukungnya Internet murah dan netbook murah, hanya saja di Indonesia operator kurang adil dalam menentukan harga dan bahkan ada salah satu operator yang sengaja membuat "jebakan" agar pengguna Internet tersebut membayar lebih mahal. Lainnya sekitar 42% dari akses Internet melalui fasilitas Public Internet Access seperti warnet , cybercafe, hotspot dll. Tempat umum lainnya yang sering dipakai untuk akses Internet adalah di kampus dan di kantor.

Disamping menggunakan PC (Personal Computer), kita juga dapat mengakses Internet melalui Handphone (HP) menggunakan fasilitas yang disebut GPRS (General Packet Radio Service). GPRS merupakan salah satu standar komunikasi wireless (nirkabel) yang memiliki kecepatan koneksi 115 kbps dan mendukung aplikasi yang lebih luas (grafis dan multimedia). Teknologi GPRS dapat diakses yang mendukung fasilitas tersebut. Pengaturan GPRS pada ponsel tergantung dari operator yang digunakan. Biaya akses Internet dihitung melalui besarnya kapasitas (per-kilobyte) yang diunduh.

### Penggunaan Internet di tempat umum

Internet juga semakin banyak digunakan di tempat umum. Beberapa tempat umum yang menyediakan layanan Internet termasuk perpustakaan, dan Internet cafe/warnet (juga disebut Cyber Cafe). Terdapat juga tempat awam yang menyediakan pusat akses Internet, seperti Internet Kiosk, Public access Terminal, dan Telepon web.

Terdapat juga toko-toko yang menyediakan akses wi-fi, seperti Wifi-cafe. Pengguna hanya perlu membawa laptop (notebook), atau PDA, yang mempunyai kemampuan wifi untuk mendapatkan akses Internet.

## Situs Web

Situs web (bahasa Inggris: web site) atau sering disingkat dengan istilah situs[1] adalah sejumlah halaman web yang memiliki topik saling terkait, terkadang disertai pula dengan berkas-berkas gambar, video, atau jenis-jenis berkas lainnya.[2] Sebuah situs web biasanya ditempatkan setidaknya pada sebuah server web yang dapat diakses melalui jaringan seperti internet, ataupun jaringan wilayah lokal (LAN) melalui alamat internet yang dikenali sebagai URL. Gabungan atas semua situs yang dapat diakses publik di internet disebut pula sebagai Waring Wera Wanua atau lebih dikenal dengan singkatan WWW. Meskipun setidaknya halaman beranda situs internet umumnya dapat diakses publik secara bebas, pada prakteknya tidak semua situs memberikan kebebasan bagi publik untuk mengaksesnya, beberapa situs web mewajibkan pengunjung untuk melakukan pendaftaran sebagai anggota, atau bahkan meminta pembayaran untuk dapat menjadi aggota untuk dapat mengakses isi yang terdapat dalam situs web tersebut, misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surel (e-mail), dan lain-lain. Pembatasan-pembatasan ini umumnya dilakukan karena alasan keamanan, menghormati privasi, atau karena tujuan komersil tertentu.

Sebuah halaman web merupakan berkas yang ditulis sebagai berkas teks biasa (plain text) yang diatur dan dikombinasikan sedemikian rupa dengan instruksi-instruksi berbasis HTML, atau XHTML, kadang-kadang pula disisipi dengan sekelumit bahasa skrip. Berkas tersebut kemudian diterjemahkan oleh peramban web dan ditampilkan seperti layaknya sebuah halaman pada monitor komputer.

Halaman-halaman web tersebut diakses oleh pengguna melalui protokol komunikasi jaringan yang disebut sebagai HTTP, sebagai tambahan untuk meningkatkan aspek keamanan dan aspek privasi yang lebih baik, situs web dapat pula mengimplementasikan mekanisme pengaksesan melalui protokol HTTPS.

### Sejarah

Penemu situs web adalah Sir Timothy John ¨Tim¨ Berners-Lee, sedangkan situs web yang tersambung dengan jaringan pertamakali muncul pada tahun 1991. Maksud dari Tim ketika merancang situs web adalah untuk memudahkan tukar menukar dan memperbarui informasi pada sesama peneliti di tempat ia bekerja. Pada tanggal 30 April 1993, CERN (tempat dimana Tim bekerja) mengumumkan bahwa WWW dapat digunakan secara gratis oleh publik.

Sebuah situs web bisa berupa hasil kerja dari perorangan atau individu, atau menunjukkan kepemilikan dari suatu organisasi, perusahaan. biasanya pembahasan dalam sebuah situs web merujuk pada sebuah ataupun beberapa topik khusus, atau kepentingan tertentu. Sebuah situs web bisa berisi pranala yang menghubungkan ke situs web lain, demkian pula dengan situs web lainnya. Hal ini terkadang membuat perbedaan antara situs web yang dibuat oleh individu ataupun perseorangan dengan situs web yang dibuat oleh organisasi bisnis menjadi tidak begitu jelas.

Situs web biasanya ditempatkan pada server web. Sebuah server web umumnya telah dilengkapi dengan perangkat-perangkat lunak khusus untuk menangani pengaturan nama ranah, serta menangani layanan atas protokol HTTP yang disebut sebagai Server HTTP (bahasa Inggris: HTTP Server) seperti Apache HTTP Server, atau Internet Information Services (IIS).

### Situs Web Statis

Situs web statis merupakan situs web yang memiliki isi tidak dimaksudkan untuk diperbarui secara berkala sehingga pengaturan ataupun pemutakhiran isi atas situs web tersebut dilakukan secara manual. Ada tiga jenis perangkat utilitas yang biasa digunakan dalam pengaturan situs web statis:

Editor teks merupakan perangkat utilitas yang digunakan untuk menyunting berkas halaman web, misalnya: Notepad atau TextEdit.

Editor WYSIWYG, merupakan perangkat lunak utilitas penyunting halaman web yang dilengkapi dengan antar muka grafis dalam perancangan serta pendisainannya, berkas halaman web umumnya tidak disunting secara lengsung oleh pengguna melainkan utilitas ini akan membuatnya secara otomatis berbasis dari laman kerja yang dibuat oleh pengguna. perangkat lunak ini misalnya: Microsoft Frontpage, Macromedia Dreamweaver.

Editor berbasis templat, beberapa utilitas tertentu seperti Rapidweaver dan iWeb, pengguna dapat dengan mudah membuat sebuah situs web tanpa harus mengetahui bahasa HTML, melainkan menyunting halaman web seperti halnya halaman biasa, pengguna dapat memilih templat yang akan digunakan oleh utilitas ini untuk menyunting berkas yang dibuat pengguna dan menjadikannya halam web secara otomatis.

### Situs Web Dinamis

Situs web dinamis merupakan situs web yang secara spesifik didisain agar isi yang terdapat dalam situs tersebut dapat diperbarui secara berkala dengan mudah. Sesuai dengan namanya, isi yang terkadung dalam situs web ini umumnya akan berubah setelah melewati satu periode tertentu. Situs berita adalah salah satu contoh jenis situs yang umumnya mengimplementasikan situs web dinamis.

Tidak seperti halnya situs web statis, pengimplementasian situs web dinamis umumnya membutuhkan keberadaan infrastruktur yang lebih kompleks dibandingkan situs web statis. Hal ini disebabkan karena pada situs web dinamis halaman web umumnya baru akan dibuat saat ada pengguna yang mengaksesnya, berbeda dengan situs web statis yang umumnya telah membentuk sejumlah halaman web saat diunggah di server web sehingga saat pengguna mengaksesnya server web hanya tinggal memberikan halaman tersebut tanpa perlu membuatnya terlebih dulu.

Untuk memungkinkan server web menciptakan halaman web pada saat pengguna mengaksesnya, umumnya pada server web dilengkapi dengan mesin penerjemah bahasa skrip (PHP, ASP, ColdFusion, atau lainnya), serta perangkat lunak sistem manajemen basisdata relasional seperti MySQL.

Struktur berkas sebuah situs web dinamis umumnya berbeda dengan situs web statis, berkas-berkas pada situs web statis umumnya merupakan sekumpulan berkas yang membentuk sebuah situs web. Berbeda halnya dengan situs web dinamis, berkas-berkas pada situs web dinamis umumnya merupakan sekumpulan berkas yang membentuk perangkat lunak aplikasi web yang akan dijalankan oleh mesin penerjemah server web, berfungsi memanajemen pembuatan halaman web saat halaman tersebut diminta oleh pengguna.

### Layanan Host

Layanan host internet adalah jasa layanan internet yang menyediakan sumber daya server-server untuk disewakan sehingga memungkinkan organisasi atau individu menempatkan informasi di internet berupa HTTP, FTP, EMAIL, atau DNS

Peladen host terdiri dari sebuah server atau gabungan server-server yang terhubung dengan jaringan internet berkecepatan tinggi.

Ada beberapa jenis layanan host yaitu host berbagi, server terdedikasi, server maya terdedikasi, dan server kolokasi.

### Perancangan Web

Perancangan web (web design) adalah istilah umum yang digunakan untuk mencakup bagaimana isi web konten ditampilkan, (biasanya berupa hypertext atau hypermedia) yang dikirimkan ke pengguna akhir melalui World Wide Web, dengan menggunakan sebuah browser web atau perangkat lunak berbasis web. Tujuan dari web design adalah untuk membuat website—sekumpulan konten online termasuk dokumen dan aplikasi yang berada pada server web / server. Sebuah website dapat berupa sekumpulan teks, gambar, suara dan konten lainnya, serta dapat bersifat interaktif ataupun statis.

**Konten**

Elemen-elemen seperti teks, forms, images (GIFs, JPEGs, Portable Network Graphics) dan video dapat diletakkan di dalam halaman menggunakan tag-tag HTML/XHTML/XML. Browser terkadang juga memerlukan Plug-ins seperti Adobe Flash, QuickTime, Java, dan sebagainya untuk menampilkan beberapa media yang diletakkan di dalam halaman web menggunakan tag-tag HTML/XHTML.

Halaman web dan situs web dapat berupa halaman statis, atau dapat diprogram secara dinamis sehingga menghasilkan halaman web dengan konten atau tampilan visual yang diinginkan, tergantung pada berbagai faktor, seperti masukan dari pengguna akhir, masukan dari Webmaster, atau perubahan dalam lingkungan komputasi (seperti situs yang terkait dengan database yang telah diubah).

* Bahasa Markup (seperti HTML, XHTML dan XML)
* Gaya lembar bahasa (seperti CSS dan XSL)
* Client-side scripting (seperti JavaScript dan VBScript)
* Server-side scripting (seperti PHP dan ASP)
* Teknologi database (seperti MySQL dan PostgreSQL)
* Teknologi multimedia (seperti Flash dan Silverlight)

**Perancang Web**

Perancang web atau desainer web (web designer) adalah orang yang memiliki keahlian menciptakan konten presentasi (biasanya hypertext atau hypermedia) yang dikirimkan ke pengguna-akhir melalui World Wide Web, menggunakan Web browser atau perangkat lunak Web-enabled lain seperti televisi internet, Microblogging, RSS, dan sebagainya.

Dengan berkembangnya spesialisasi dalam desain komunikasi dan bidang teknologi informasi, ada kecenderungan kuat untuk menarik garis yang jelas antara web design khusus untuk halaman web dan pengembangan web secara keseluruhan dari semua layanan berbasis web.

## PHP (PHP Hypertext Preprocessor)

PHP: Hypertext Preprocessor adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memrogram situs web dinamis. PHP dapat digunakan untuk membangun sebuah CMS.

### Sejarah

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

### Kelebihan PHP dari bahasa pemrograman lain

Beberapa kelebihan PHP dari bahasa pemrograman web, antara lain:

'Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaanya.'

'Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relatif mudah.'

'Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.'

'Dalam sisi pemahamanan, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.'

'PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah system.'

### Tipe data

PHP memiliki 8 (delapan) tipe data yaitu :

1. Integer
2. Double
3. Boolean
4. String
5. Object
6. Array
7. Null
8. Nill
9. Resource

## Yii Framework

Yii adalah sebuah framework performa tinggi PHP yang terbaik untuk mengembangkan aplikasi web 2.0.

**Yii membantu pengembang web membangun aplikasi kompleks dan memberikan nya tepat waktu.**

Yii dibaca Yee atau [ji:], dan adalah singkatan untuk “***Yes It Is****”.* Biasanya yang akurat, dan respon paling jelas untuk pertanyaan dari siapa saja yang bari belajar yii:

*“Is it fast? … Is it Secure? … Is it profesional? … Is it right for my next project? ...* ***Yest, it is!***

Yii gratis, framework aplikasi pengembangan web *open-source* yang ditulis dalam PHP5 yang mempromosikan bersih, desain matang dan mendorong pengembangan secara cepat. Bekerja untuk mempersingkat pengembangan aplikasi web dan membantu untuk meyakinkan sebuah produk yang sangat efisien, dapat diperluas, dan di perbaharui.

## Netbeans IDE

**NetBeans** mengacu pada dua hal, yakni platform untuk pengembangan aplikasi desktop java, dan sebuah *Integrated Development Environment*(IDE) yang dibangun menggunakan platform NetBeans.

Platform NetBeans memungkinkan aplikasi dibangun dari sekumpulan komponen perangkat lunak moduler yang disebut ‘modul’. Sebuah modul adalah suatu arsip Java (Java *archive*) yang memuat kelas-kelas Java untuk berinetraksi dengan *NetBeans Open API* dan file manifestasi yang mengidentifikasinya sebagai modul.

Karena modul dapat dikembangkan secara independen, aplikasi berbasis platform NetBeans dapat dengan mudah dikembangkan oleh pihak ketiga secara mudah dan *powerful*.

# ANALISIS KEBUTUHAN

Sistem web sekolah yang akan dikembangkan adalah sistem banyak pengguna (*multi-user system)* dimana pengguna nya kira-kira adalah Admin, Operator, Guru, dan Siswa. Berikut adalah skenario yang dapat dilakukan oleh setiap user:

Admin dapat melakukan tindakan sebagai berikut:

* Login dan Logout
* Melakukan semua tindakan termasuk manajemen pengguna
* Melihat Log

Guru dapat melakukan tindakan sebagai berikut:

* Login dan Logout
* Membuat, Mengubah, Menghapus berita
* Membuat, Mengubah, Menghapus agenda
* Membuat, Mengubah, Menghapus dan mengatur polling
* Membuat, Mengubah, Menghapus pengumuman
* Menyetujui komentar
* Mengatur kalender akademik
* Mengatur highligh berita
* Melihat Log

Operator dapat melakukan tindakan sebagai berikut:

* Membuat, mengubah, menghapus berita
* Membuat, mengubah, menghapus pada item gallery
* Membuat, mengubah, menghapus, dan mengatur mading
* Mengatur highlight berita

Siswa dapat melakukan tindakan sebagai berikut:

* Login dan Logout
* Membaca berita, agenda, pengumuman
* Mengakses halaman direktori siswa dan guru
* Membuat komentar pada berita
* Melihat Gallery, Mading, hasil polling, dan kalender akademik
* Memberikan polling
* Membuat pesan pada buku tamu

Pengguna yang lain yang merupakan tamu dapat melakukan tindakan sebagai berikut:

* Membaca berita,agenda, dan pengumuman
* Melihat gallery, hasil polling, Mading, Kalender akademik
* Menulis buku tamu

Kebutuhan tambahan untuk sistem ini yaitu:

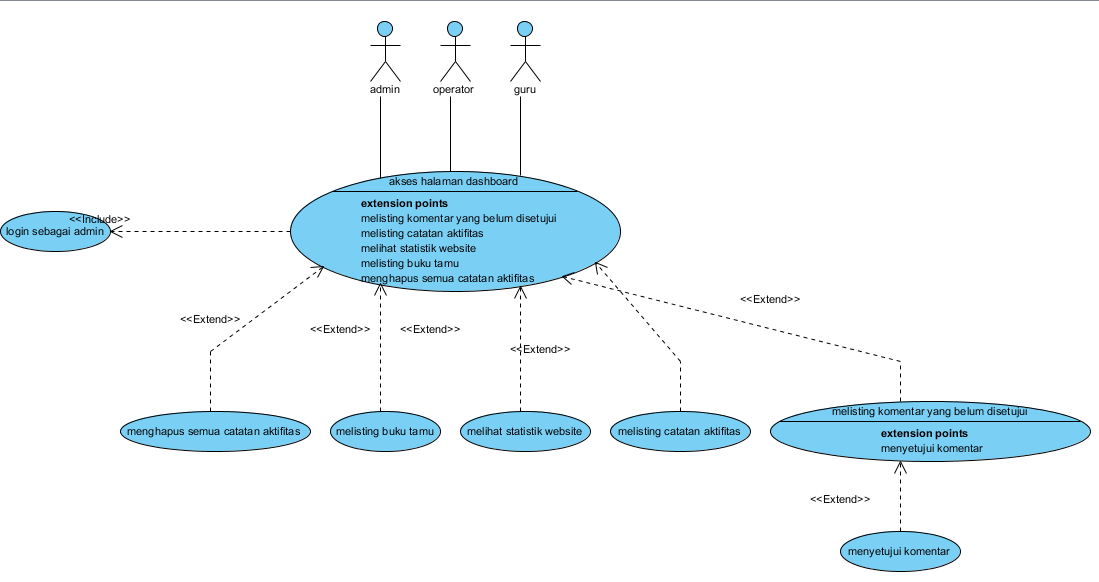
* Sistem
  + Sistem menggunakan *SEO-friendly URLs*
  + Sistem mempunyai tema yang dapat diubah
  + Sistem dapat menampilkan komentar paling baru
  + Sistem dapat menampilkan berita sesuai dengan tag nya
  + Sistem dapat menampilkan berita bersamaan dengan komentar
  + Sistem dapat menampilkan kalender akademik disetiap halaman
* Halaman depan
  + menampilkan highlight berita berupa slideshow dengan jumlah maksimal 5 berita.
  + Menampilkan agenda
  + Menampilkan pengumuman
  + Menampilkan statistik website

## Use Case Back-End

Pada halaman backend hanya user yang mempunyai hak akses admin, guru, dan operator yang dapat mengakses halaman admin. user akan dipersilahkan untuk login saat akan memasuki halaman admin.

### Dashboard

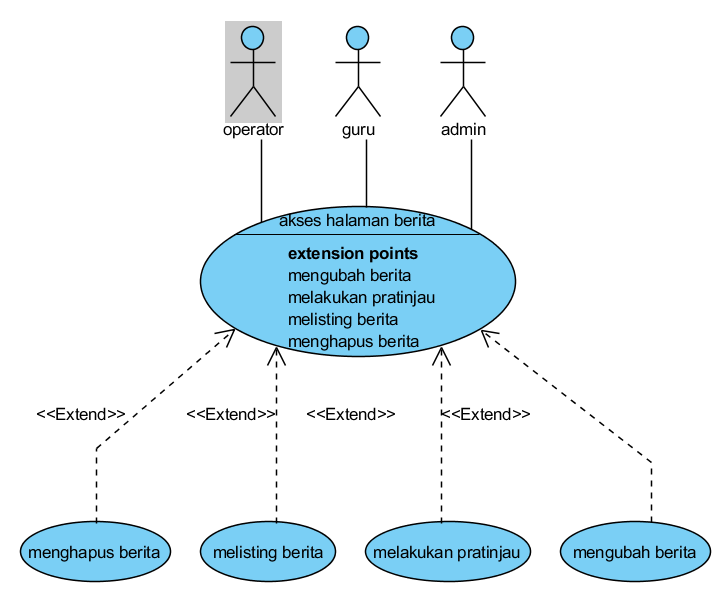
Halaman dashboard, yaitu halaman utama admin. Pada halaman tersebut admin, guru, dan operator dapat melihat statistik website, melisting komentar yang belum disetujui, melisting buku tamu, melisting catatan aktifitas. Mereka juga dapat menyetujui komentar yang belum disetujui. Admin dapat menghapus semua catatan aktifitas.



Gambar 3.1 Perancangan use case Back-End Dashboard

### Berita

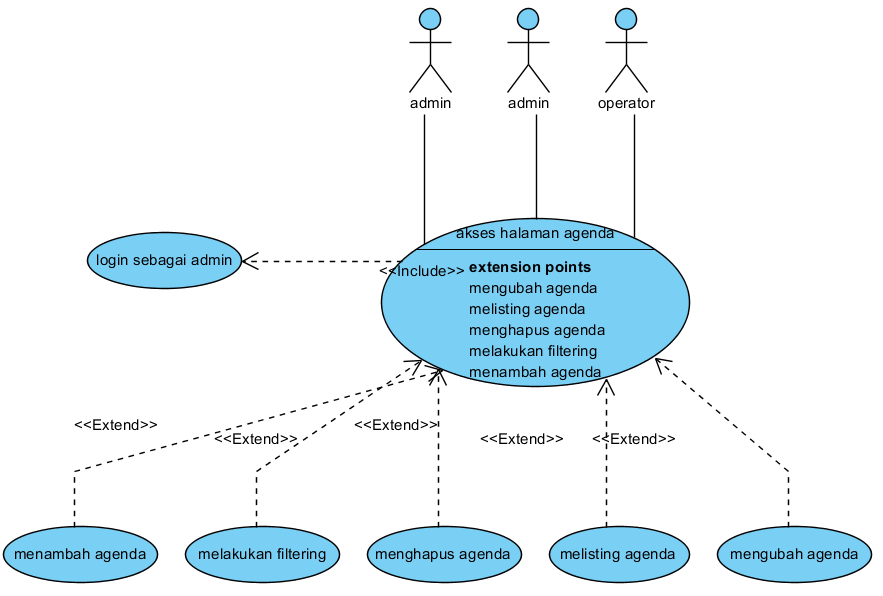
Halaman berita, yaitu halaman untuk mengatur berita. Pada halaman tersebut admin, guru dan operator dapat melisting berita,menambah berita, menghapus berita,mengubah berita, melakukan pratinjau pada berita dan juga dapat melakukan filtering.



Gambar 3.2 use case Back-End Berita

### Agenda

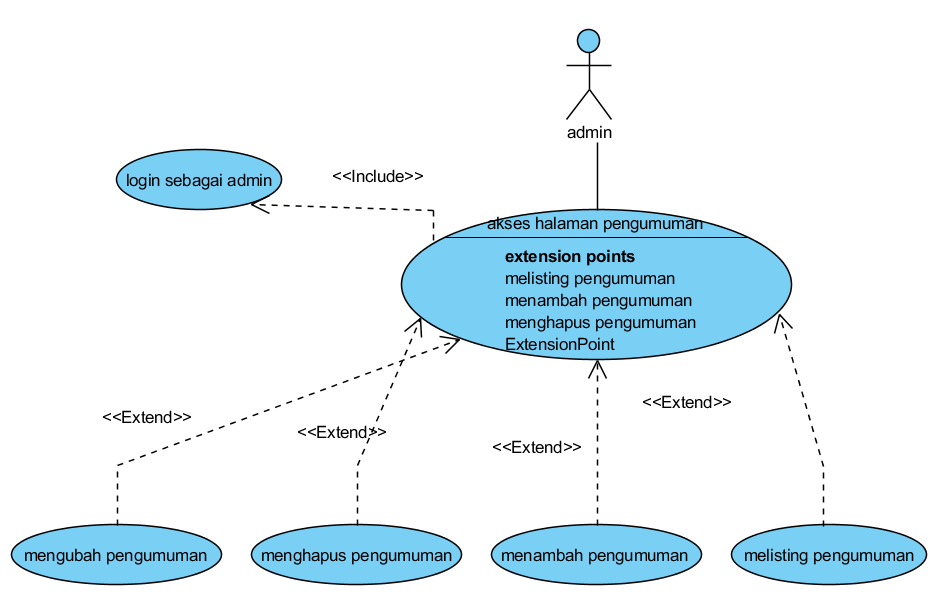
Halaman agenda, yaitu halaman untuk mengatur agenda. Pada halaman tersebut admin dan guru, dapat melisting agenda, menambah agenda, menghapus agenda, mengubah agenda, dan juga dapat melakukan filtering.



Gambar 3.3 Use Case Back-End Agenda

### Pengumuman

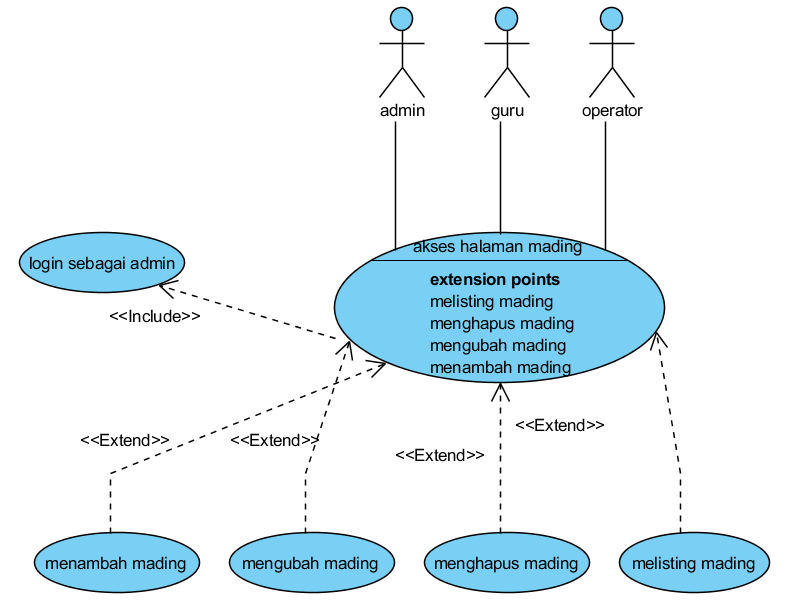
Halaman pengumuman, yaitu halaman untuk mengatur dan mengorganisir pengumuman. Pada halaman tersebut admin dan guru dapat melisting pengumuman, menambah pengumuman, menghapus pengumuman, dan mengubah pengumuman.



Gambar 3.4 Use Case Back-End Pengumuman

### Mading

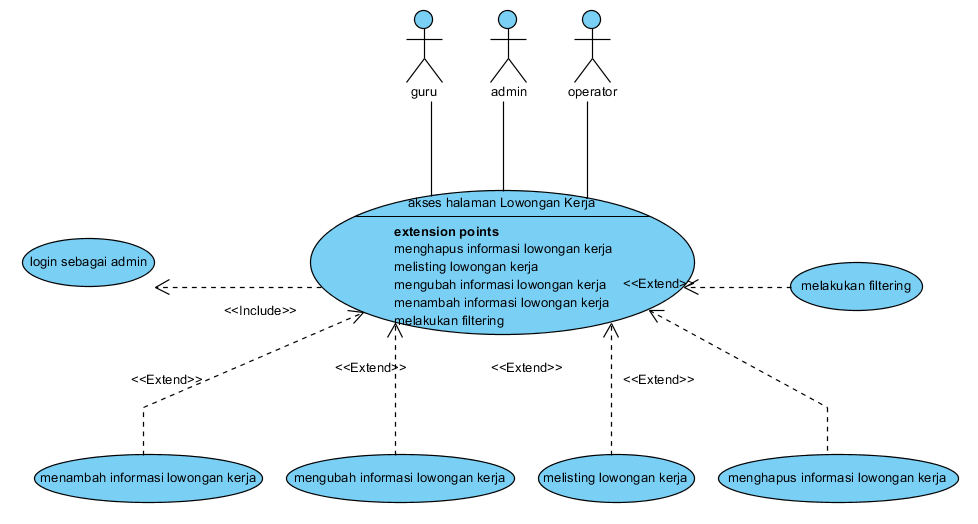
Halaman mading, yaitu halaman untuk mengatur dan mengorganisir mading. Pada halaman tersebut admin, guru, dan operator dapat melisting mading, menambah mading, menghapus mading, dan mengubah mading.



Gambar 3.5 Use Case Back-End Agenda

### Informasi Lowongan Kerja

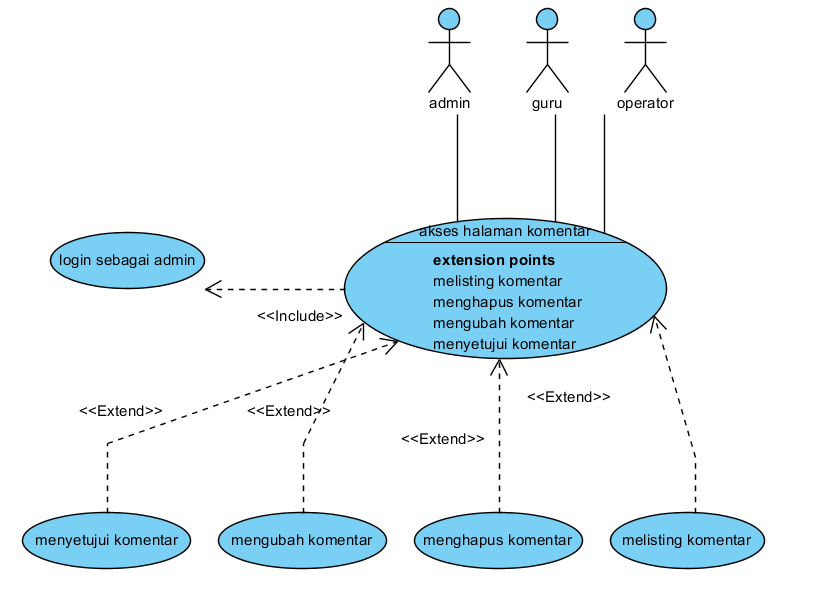
Lowongan Kerja, yaitu halaman untuk mengatur informasi lowongan kerja. pada halaman tersebut admin, guru, dan operator dapat melisting lowongan kerja, menambah informasi lowongan kerja, menghapus informasi lowongan kerja, dan mengubah informasi lowongan kerja. Mereka juga dapat melakukan filtering.



Gambar 3.6 Use Case Back-End Lowongan Kerja

### Komentar

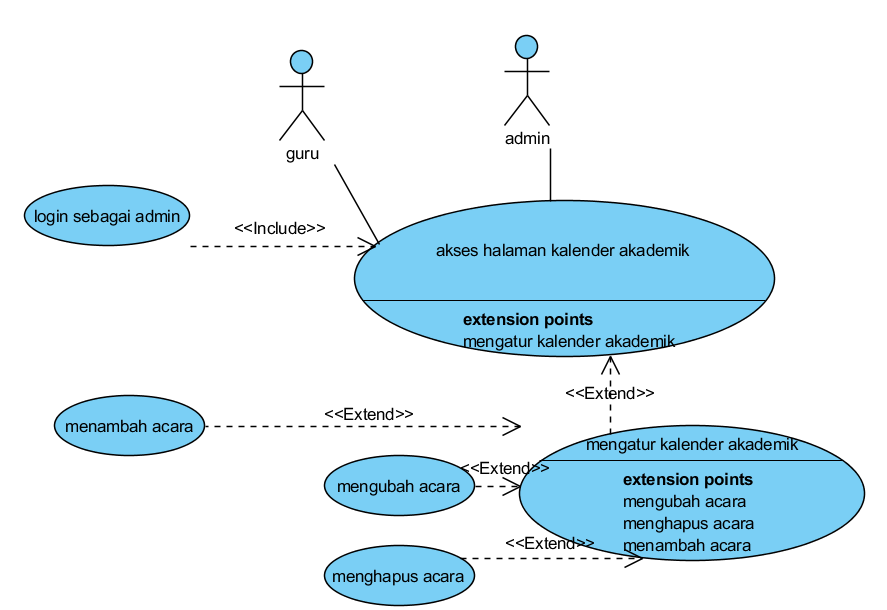
Halaman komentar. yaitu halaman untuk mengorganisir komentar. pada halaman tersebut admin, guru, dan operator dapat melisting komentar, menghapus komentar, mengubah komentar, dan menyetujui komentar yang belum disetujui.



Gambar 3.7 Use Case Back-End Komentar

### Kalender Akademik

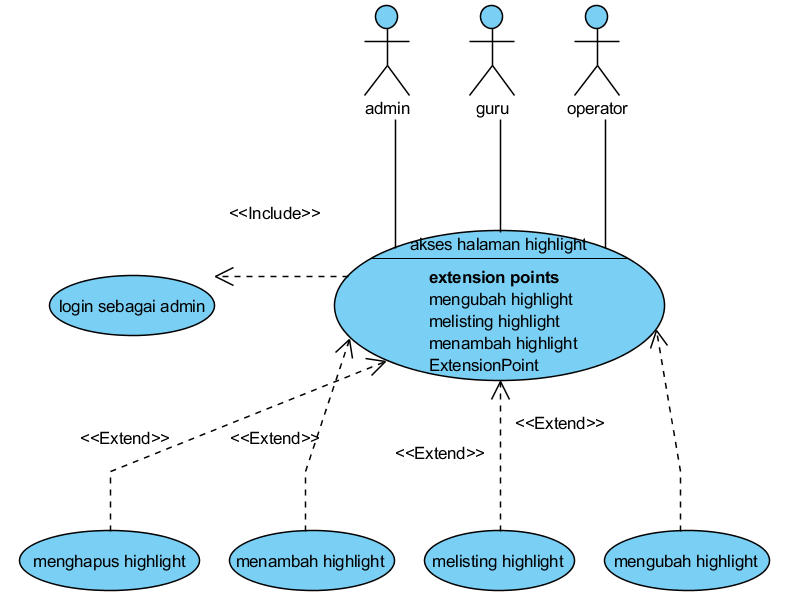
Halaman kalender akademik, yaitu halaman untuk mengorganisir kalender akademik. Pada halaman tersebut admin dan guru dapat melihat kalender akademik dan mengatur kalender akademik seperti menambah acara kedalam kalender akademik, menghapus acara dari kalender akademik, dan mengubah acara.



Gambar 3.8 Use Case Back-End Kalender Akademik

### Highlight

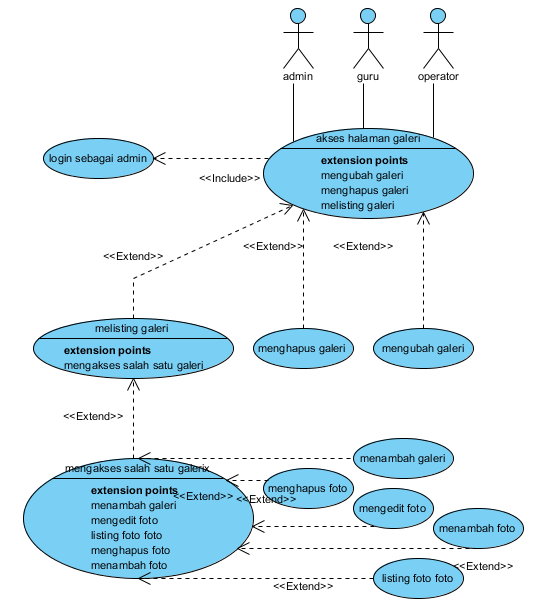
Halaman highlight, yaitu halaman untuk mengorganisir highlight. Pada halaman tersebut admin, guru, dan operator dapat melisting highlight dan dapat menambah highlight, menghapus highlight, dan mengubah highlight.



Gambar 3.9 Use Case Back-End Highlight

### Galeri

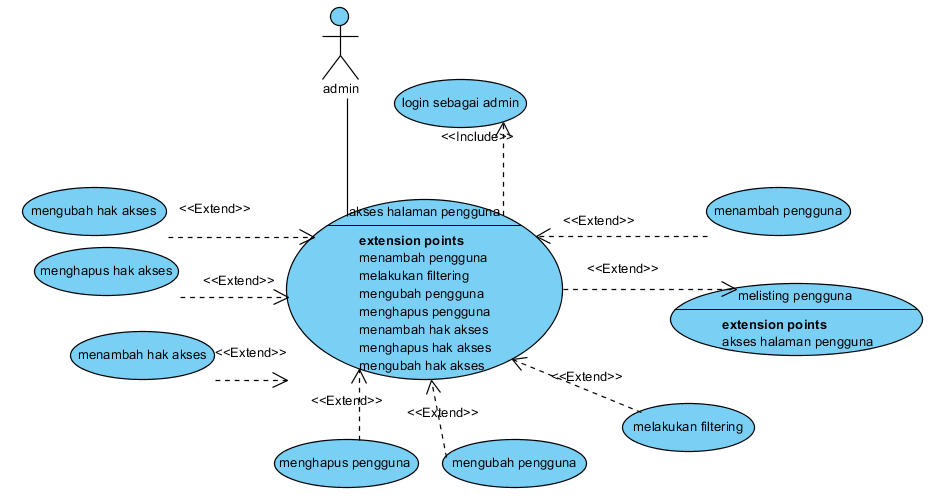
Halaman galeri, yaitu halaman untuk mengorganisir galeri. Pada halaman tersebut admin, guru, dan operator dapat melisting galeri seperti halnya folder. mereka juga dapat menambah galeri, menghapus galeri, dan mengubah galeri. mereka dapat mengakses salah satu galeri dan melihat listing foto foto, dan mereka juga dapat menambah foto, menghapus foto, dan mengedit foto.



Gambar 3.10 Use Case Back-End Galeri

### Pengguna

Halaman pengguna, yaitu halaman untuk mengatur pengguna. pada halaman tersebut admin dapat melisting pengguna, menambah pengguna, menghapus pengguna, mengubah pengguna dan juga melakukan filtering. Pada halaman itu juga admin dapat mengorganisir hak akses seperti menambah hak akses, menghapus hak akses, dan mengubah hak akses.



Gambar 3.11 Use Case Back-End Pengguna

## Analisis Kebutuhan Sistem

Teknologi yang digunakan untuk menjalankan aplikasi dari segi kebutuhan hardware dan software adalah sebagai berikut :

### Spesifikasi Software

Sistem Operasi : Windows XP Professional SP2 ke atas, Window 7

Bahasa Pemrograman : Java

Editor : NetBeans IDE 7.1

Platform : J2ME (*Java 2 Micro Editin)*

Emulator : Java™ Platform Micro Edition SDK 3.0

### Spesifikasi Hardware

Prosesor : Intel® PentiumIV® CPU 2.0Ghz

Memory : 512 MB

Harddisk : 300 MB

Graphic Card : 64 MB

### Spesifikasi Handphone

Fitur : JAVA

Layar : 240 x 320 px

Device Configuration : CLDC 1.1

Device Profile : MIDP-2.1

# PERANCANGAN

## Perancangan Tampilan

Perancangan tampilan merupakan rancangan tampilan form yang digunakan sebelum permainan “Si Kancil” di mulai, seperti tampilan menu utama, petunjuk dan tentang.

### Tampilan *Splash Screen*

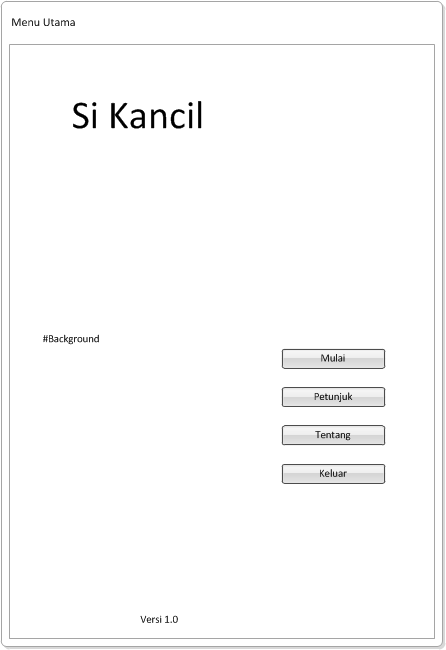
Tampilan *Splash Screen* merupakan tampilan saat pertamakali aplikasi dimulai, berisi tentang judul dari aplikasi yang akan dijalankan. Adapun tampilan *Splash Screen* pada Game “Si Kancil” Sebagai berikut :



Gambar 4.1 Perancangan tampilan *splash screen*

### Perancangan tampilan menu utama

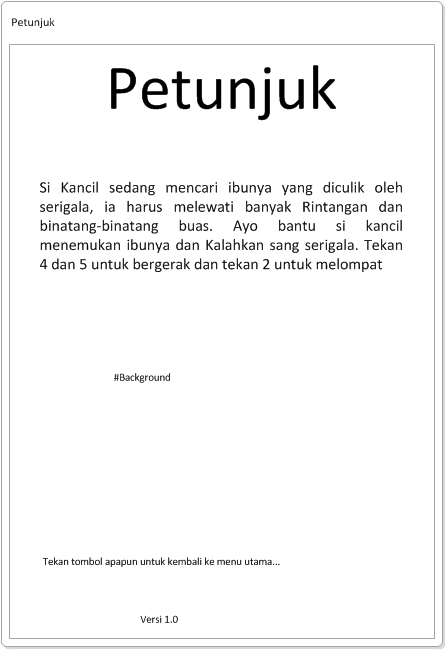
Menu utama merupakan tampilan setelah *splash screen* selesai. Menu utama berguna untuk navigasi aplikasi dan merupakan pilihan pilihan sebelum aplikasi dimulai. Adapun menu utama pada Game “Si Kancil” adalah sebagai berikut :



Gambar 4.2 Perancangan tampilan menu utama

### Perancangan tampilan petunjuk

Tampilan petunjuk merupakan tampilan untuk petunjuk aplikasi meliputi cara menggunakan aplikasi dan tombol tombol yang digunakan. Adapun perancangan tampilan tampilan petunjuk yaitu sebagai berikut :



Gambar 4.3 Perancangan tampilan petunjuk

### Perancangan tampilan tentang

Tampilan tentang merupakan tampilan yang berisi tentang aplikasi meliputi Judul aplikasi, versi, nama pembuat, serta kontak pembuat bila perlu. Adapun perancangan tampilan tentang pada Game “Si Kancil” yaitu sebagai berikut :



Gambar 4.4 Perancangan tampilan tentang

### Perancangan tampilan “*Game Over”*

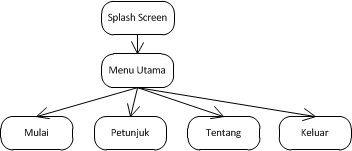
Tampilan “*Game Over”* Merupakan tampilan saat pemain kalah dari permainan. Berikut tampilan “*Game Over”* pada Game “Si Kancil”:



Gambar 4.5 Perancangan tampilan “*Game Over”*

## Gambaran Struktur Tampilan

Gambaran struktur tampilan yang meliputi keseluruhan tampilan pada Game “Si Kancil”



Gambar 4.6 Perancangan struktur tampilan

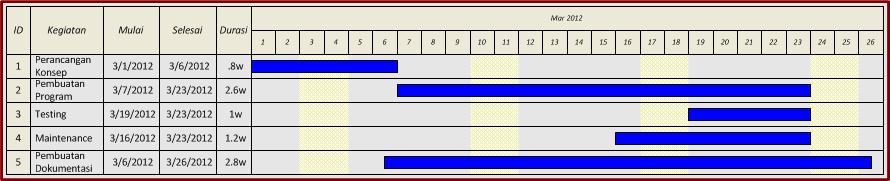
## Jadwal Kerja

Perencanaan jadwal kerja pada pembuata Game “Si Kancil”

### Gantt Chart

*Gantt chart* adalah suatu alat yang bernilai khususnya untuk proyek-proyek dengan jumlah anggota tim yang sedikit, proyek mendekati penyelesaian dan beberapa kendala proyek.

Tabel 4.1 Gantt Chart



# IMPLEMENTASI

## Implementasi Tampilan

Implementasi dari perancangan tampilan yang sudah dibuat pada bab perancangan

### Tampilan *Splash Screen*



Gambar 5.1 Implementasi tampilan *splash screen*

|  |
| --- |
| MidletSiKancil.java |
| /\*  \* To change this template, choose Tools | Templates  \* and open the template in the editor.  \*/  import java.io.IOException;  import javax.microedition.lcdui.Display;  import javax.microedition.lcdui.Image;  import javax.microedition.midlet.MIDlet;  import org.netbeans.microedition.lcdui.SplashScreen;  /\*\*  \* @author egijelek  \*/  public class MidletSiKancil extends MIDlet {  private GameCanvasSiKancil canvas;  private GameOverCanvas gameOverCanvas;  private Petunjuk petunjuk;  private Tentang tentang;  private MainMenu mainMenu;  private Thread t;  private Display display;  private SplashScreen splash;  public void startApp() {  try {    display = Display.getDisplay(this);  mainMenu = new MainMenu(this);  tentang = new Tentang(this);  petunjuk = new Petunjuk(this);  splash = new SplashScreen(display);  splash.setTimeout(3000);  splash.setImage(Image.createImage(getClass().getResourceAsStream("/res/splash.png")));  splash.setFullScreenMode(true);  splash.setNextDisplayable(mainMenu);  display.setCurrent(splash);  } catch (Exception ex) {  ex.printStackTrace();  }  }  public Display getDisplay() {  return display;  }  public void pauseApp() {  }  public void destroyApp(boolean unconditional) {  }  public void gameOver(int score) {  try {  gameOverCanvas = new GameOverCanvas(score, this);  display.setCurrent(gameOverCanvas);  } catch (IOException ex) {  ex.printStackTrace();  }  }  public void newGame() {  try {  canvas = new GameCanvasSiKancil(this);  t = new Thread(canvas);  t.start();  display.setCurrent(canvas);  } catch (Exception ex) {  ex.printStackTrace();  }  }  void petunjuk() {  display.setCurrent(petunjuk);  }  void tentang() {  display.setCurrent(tentang);  }  public void mainMenu() {  display.setCurrent(mainMenu);  }  } |

### Tampilan Menu Utama



Gambar 5.2 Implementasi tampilan menu utama

|  |
| --- |
| MainMenu.java |
| import java.io.IOException;  import javax.microedition.lcdui.\*;  import javax.microedition.media.Manager;  import javax.microedition.media.MediaException;  import javax.microedition.media.Player;  /\*\*  \*  \* @author egijelek  \*/  public class MainMenu extends Canvas {  private GameCanvasSiKancil canvasSiKancil;  private MidletSiKancil midlet;  private Image background;  private String[] menu = {"Mulai", "Petunjuk", "Tentang", "Keluar"};  private int selection = 0;  private Player playerSelectionEffect;  public MainMenu(MidletSiKancil midlet) throws IOException, MediaException {  setFullScreenMode(true);  this.midlet = midlet;  background = Image.createImage(getClass().getResourceAsStream("/res/MainMenu.png"));  playerSelectionEffect = Manager.createPlayer(getClass().getResourceAsStream("/res/fruit.wav"), "audio/wav");  }  protected void paint(Graphics g) {  g.drawImage(background, getWidth(), getHeight(), Graphics.BOTTOM | Graphics.RIGHT);  g.setFont(Font.getFont(Font.FACE\_PROPORTIONAL, Font.STYLE\_BOLD, Font.SIZE\_LARGE));  for (int i = 0; i < menu.length; i++) {  if (selection == i) {  g.setColor(255, 0, 0);  g.drawString(menu[i], getWidth() - (getWidth() / 3), getHeight() / 2 + (15 \* (i)), Graphics.TOP | Graphics.LEFT);  } else {  g.setColor(0, 0, 0);  g.drawString(menu[i], getWidth() - (getWidth() / 3), getHeight() / 2 + (15 \* (i)), Graphics.TOP | Graphics.LEFT);  }  }  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  switch (getGameAction(keyCode)) {  case UP:  if (selection == 0) {  return;  }  try {  playerSelectionEffect.start();  } catch (MediaException ex) {  ex.printStackTrace();  }  selection--;  repaint();  break;  case DOWN:  if (selection == menu.length - 1) {  return;  }  try {  playerSelectionEffect.start();  } catch (MediaException ex) {  ex.printStackTrace();  }  selection++;  repaint();  break;  case FIRE:  switch (selection) {  case 0:  midlet.newGame();break;  case 1:  midlet.petunjuk();break;  case 2:  midlet.tentang();break;  case 3:  midlet.notifyDestroyed();  }  }  }  } |

### Tampilan Petunjuk



Gambar 5.3 Implementasi tampilan petunjuk

|  |
| --- |
| Petunjuk.java |
| import java.io.IOException;  import javax.microedition.lcdui.Canvas;  import javax.microedition.lcdui.Graphics;  import javax.microedition.lcdui.Image;  /\*\*  \*  \* @author egijelek  \*/  public class Petunjuk extends Canvas{  private Image petunjuk;  private MidletSiKancil midlet;    public Petunjuk(MidletSiKancil midlet) throws IOException {  setFullScreenMode(true);  petunjuk = Image.createImage(getClass().getResourceAsStream("/res/petunjuk.png"));  this.midlet = midlet;    }  protected void paint(Graphics g) {  g.drawImage(petunjuk, 0, 0, Graphics.TOP | Graphics.LEFT);  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  midlet.mainMenu();    }  } |

### Tampilan Tentang



Gambar 5.4 Implementasi tampilan tentang

|  |
| --- |
| Tentang.java |
| import java.io.IOException;  import javax.microedition.lcdui.Canvas;  import javax.microedition.lcdui.Graphics;  import javax.microedition.lcdui.Image;  /\*\*  \*  \* @author egijelek  \*/  public class Tentang extends Canvas{  private Image petunjuk;  private MidletSiKancil midlet;    public Tentang(MidletSiKancil midlet) throws IOException {  setFullScreenMode(true);  petunjuk = Image.createImage(getClass().getResourceAsStream("/res/tentang.png"));  this.midlet = midlet;    }  protected void paint(Graphics g) {  g.drawImage(petunjuk, 0, 0, Graphics.TOP | Graphics.LEFT);  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  midlet.mainMenu();    }  } |

### Tampilan Game Over



Gambar 5.5 Implementasi tampilan game over

|  |
| --- |
| GameOverCanvas.java |
| import java.io.IOException;  import javax.microedition.lcdui.Canvas;  import javax.microedition.lcdui.Graphics;  import javax.microedition.lcdui.Image;  /\*\*  \*  \* @author egijelek  \*/  public class GameOverCanvas extends Canvas{    private int score;  private Image background;  private MidletSiKancil midlet;  public GameOverCanvas (int score, MidletSiKancil midletSiKancil) throws IOException{  this.score = score;  this.midlet = midletSiKancil;  background = Image.createImage(getClass().getResourceAsStream("/res/GameOver.png"));  }  protected void paint(Graphics g) {  g.drawImage(background, 0, 0, Graphics.TOP | Graphics.LEFT);  g.drawString("Skor = " +score+"", getWidth()/2-15, getHeight()/2, Graphics.TOP | Graphics.LEFT);  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  midlet.mainMenu();  }  } |

## Implementasi Game

Implementasi pada saat game dimulai, berikut adalah contoh tampilan pada saat game “Si Kancil” Dimulai



Gambar 5.6 Implementasi game “Si Kancil”

# PENUTUP

## Kesimpulan

Game “Si Kancil” berbasis J2ME dibuat yang bertujuan sebagai hiburan khususnya kalangan kanak-kanak. Dengan bertemakan cerita “Si Kancil” diharapkan banyak pengguna yang tertarik untuk memainkannya hanya untuk mengisi waktu istirahat nya.

## Saran

Game “Si Kancil” masih dalam tahap pengembangan yang didalamnya masih banyak kekurangan seperti binatang binatang nya belum lengkap dan level nya masih sedikit sehingga perlu pengembangan

# DAFTAR PUSTAKA

Dr. Onno W. Purbo,sekilas-java , <http://onno.vlsm.org/v11/ref-ind-1/network/sekilas-java-1997.rtf> (10 November 2011 - Pkl 10:13 WIB).

Avestro, Joyce and Team. 2007. *Jeni1/Pengenalan Pemrograman 1/Mengenali Lingkup Pemrograman Anda*. Malang: JEDI

Dian, M. 2008, Pemrograman J2ME, melalui <http://www.ilmu>komputer.com/[16:02 pm des 20 2011]

FARIS , AGUNG FAISHAL .2010. *Cerita Si Kancil*.Bandung

LAMPIRAN A  
SOURCE CODE

A.1 MidletSiKancil.java

|  |
| --- |
| import java.io.IOException;  import javax.microedition.lcdui.Display;  import javax.microedition.lcdui.Image;  import javax.microedition.midlet.MIDlet;  import org.netbeans.microedition.lcdui.SplashScreen;  /\*\*  \* @author egijelek  \*/  public class MidletSiKancil extends MIDlet {  private GameCanvasSiKancil canvas;  private GameOverCanvas gameOverCanvas;  private Petunjuk petunjuk;  private Tentang tentang;  private MainMenu mainMenu;  private Thread t;  private Display display;  private SplashScreen splash;  public void startApp() {  try {    display = Display.getDisplay(this);  mainMenu = new MainMenu(this);  tentang = new Tentang(this);  petunjuk = new Petunjuk(this);  splash = new SplashScreen(display);  splash.setTimeout(3000);  splash.setImage(Image.createImage(getClass().getResourceAsStream("/res/splash.png")));  splash.setFullScreenMode(true);  splash.setNextDisplayable(mainMenu);  display.setCurrent(splash);  } catch (Exception ex) {  ex.printStackTrace();  }  }  public Display getDisplay() {  return display;  }  public void pauseApp() {  }  public void destroyApp(boolean unconditional) {  }  public void gameOver(int score) {  try {  gameOverCanvas = new GameOverCanvas(score, this);  display.setCurrent(gameOverCanvas);  } catch (IOException ex) {  ex.printStackTrace();  }  }  public void newGame() {  try {  canvas = new GameCanvasSiKancil(this);  t = new Thread(canvas);  t.start();  display.setCurrent(canvas);  } catch (Exception ex) {  ex.printStackTrace();  }  }  void petunjuk() {  display.setCurrent(petunjuk);  }  void tentang() {  display.setCurrent(tentang);  }  public void mainMenu() {  display.setCurrent(mainMenu);  }  } |

A.2 MainMenu.java

|  |
| --- |
| import java.io.IOException;  import javax.microedition.lcdui.\*;  import javax.microedition.media.Manager;  import javax.microedition.media.MediaException;  import javax.microedition.media.Player;  /\*\*  \*  \* @author egijelek  \*/  public class MainMenu extends Canvas {  private GameCanvasSiKancil canvasSiKancil;  private MidletSiKancil midlet;  private Image background;  private String[] menu = {"Mulai", "Petunjuk", "Tentang", "Keluar"};  private int selection = 0;  private Player playerSelectionEffect;  public MainMenu(MidletSiKancil midlet) throws IOException, MediaException {  setFullScreenMode(true);  this.midlet = midlet;  background = Image.createImage(getClass().getResourceAsStream("/res/MainMenu.png"));  playerSelectionEffect = Manager.createPlayer(getClass().getResourceAsStream("/res/fruit.wav"), "audio/wav");  }  protected void paint(Graphics g) {  g.drawImage(background, getWidth(), getHeight(), Graphics.BOTTOM | Graphics.RIGHT);  g.setFont(Font.getFont(Font.FACE\_PROPORTIONAL, Font.STYLE\_BOLD, Font.SIZE\_LARGE));  for (int i = 0; i < menu.length; i++) {  if (selection == i) {  g.setColor(255, 0, 0);  g.drawString(menu[i], getWidth() - (getWidth() / 3), getHeight() / 2 + (15 \* (i)), Graphics.TOP | Graphics.LEFT);  } else {  g.setColor(0, 0, 0);  g.drawString(menu[i], getWidth() - (getWidth() / 3), getHeight() / 2 + (15 \* (i)), Graphics.TOP | Graphics.LEFT);  }  }  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  switch (getGameAction(keyCode)) {  case UP:  if (selection == 0) {  return;  }  try {  playerSelectionEffect.start();  } catch (MediaException ex) {  ex.printStackTrace();  }  selection--;  repaint();  break;  case DOWN:  if (selection == menu.length - 1) {  return;  }  try {  playerSelectionEffect.start();  } catch (MediaException ex) {  ex.printStackTrace();  }  selection++;  repaint();  break;  case FIRE:  switch (selection) {  case 0:  midlet.newGame();break;  case 1:  midlet.petunjuk();break;  case 2:  midlet.tentang();break;  case 3:  midlet.notifyDestroyed();  }  }  }  } |

A.3 Petunjuk.java

|  |
| --- |
| import java.io.IOException;  import javax.microedition.lcdui.Canvas;  import javax.microedition.lcdui.Graphics;  import javax.microedition.lcdui.Image;  /\*\*  \*  \* @author egijelek  \*/  public class Petunjuk extends Canvas{  private Image petunjuk;  private MidletSiKancil midlet;    public Petunjuk(MidletSiKancil midlet) throws IOException {  setFullScreenMode(true);  petunjuk = Image.createImage(getClass().getResourceAsStream("/res/petunjuk.png"));  this.midlet = midlet;    }  protected void paint(Graphics g) {  g.drawImage(petunjuk, 0, 0, Graphics.TOP | Graphics.LEFT);  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  midlet.mainMenu();    }  } |

A.4 Tentang.java

|  |
| --- |
| import java.io.IOException;  import javax.microedition.lcdui.Canvas;  import javax.microedition.lcdui.Graphics;  import javax.microedition.lcdui.Image;  /\*\*  \*  \* @author egijelek  \*/  public class Tentang extends Canvas{  private Image petunjuk;  private MidletSiKancil midlet;    public Tentang(MidletSiKancil midlet) throws IOException {  setFullScreenMode(true);  petunjuk = Image.createImage(getClass().getResourceAsStream("/res/tentang.png"));  this.midlet = midlet;    }  protected void paint(Graphics g) {  g.drawImage(petunjuk, 0, 0, Graphics.TOP | Graphics.LEFT);  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  midlet.mainMenu();    }  } |

A.5 GameCanvasSiKancil.java

|  |
| --- |
| import java.util.Timer;  import java.util.TimerTask;  import javax.microedition.lcdui.\*;  import javax.microedition.lcdui.game.GameCanvas;  import javax.microedition.lcdui.game.LayerManager;  import javax.microedition.lcdui.game.Sprite;  import javax.microedition.lcdui.game.TiledLayer;  import javax.microedition.media.Manager;  import javax.microedition.media.MediaException;  import javax.microedition.media.Player;  /\*\*  \*  \* @author egijelek  \*/  public class GameCanvasSiKancil extends GameCanvas implements Runnable {  private static final int GRAVITY = 8;  private boolean interrupted;  private int lastDirection = RIGHT;  private int speed = 3;  private int jump = 0;  private boolean invulnerable;  private int viewPortX = 0;  private int viewPortY = 0;  private int fruits = 0;  private int live = 4;  private Image imageHeart, imageFruit;  private Player playerBackgroundSound, playerBonusEffect;  private GameDesign gd;  private Sprite spriteKancil, spriteIbuKancil;  private Sprite[] spritesEnemy, spritesAnimal;  private SpriteAnimator spriteKancilAnimator, spriteIbuKancilAnimator;  private SpritesRandomMovement spritesEnemyRandomMovement, spritesAnimalRandomMovement;  private LayerManager lm;  private TiledLayer tlPath, tlBackground, tlTrunk, tlBonus, tlGround;  private TileAnimator tlGroundWaterAnimator, tlGroundWaterRocksAnimator;  private Timer t;  private Timer tInvulnerable;  private MidletSiKancil midlet;  public GameCanvasSiKancil(MidletSiKancil midlet) throws Exception {  super(true);  setFullScreenMode(true);  this.midlet = midlet;  imageHeart = Image.createImage(getClass().getResourceAsStream("/res/heart.png"));  imageFruit = Image.createImage(getClass().getResourceAsStream("/res/fruit.png"));  playerBackgroundSound = Manager.createPlayer(getClass().getResourceAsStream("/res/kancil.mid"), "audio/midi");  playerBonusEffect = Manager.createPlayer(getClass().getResourceAsStream("/res/fruit.wav"), "audio/wav");  gd = new GameDesign();  lm = new LayerManager();  t = new Timer();  tInvulnerable = new Timer();  spriteKancil = gd.getSpriteKancil();  spriteIbuKancil = gd.getSpriteIbuKancil();  spritesEnemy = gd.getSpritesEnemy();  spritesAnimal = gd.getSpritesAnimal();  tlPath = gd.getLevel1\_path();  tlBackground = gd.getLevel1\_bg();  tlTrunk = gd.getLevel1\_trunks();  tlBonus = gd.getLevel1\_bonus();  tlGround = gd.getLevel1\_ground();  spriteIbuKancilAnimator = new SpriteAnimator(spriteIbuKancil, true);  spritesEnemyRandomMovement = new SpritesRandomMovement(spritesEnemy, gd.enemiesSpeed, gd.enemiesDelay, gd.enemiesDirections, tlPath, spriteKancil, false);  spritesAnimalRandomMovement = new SpritesRandomMovement(spritesAnimal, gd.animalsSpeed, gd.animalsDelay, gd.animalsDirection, tlPath, spriteKancil, true);  spriteKancilAnimator = new SpriteAnimator(spriteKancil, false);  tlGroundWaterAnimator = new TileAnimator(tlGround, gd.waterLevel1\_ground, gd.waterseq001, true);  tlGroundWaterRocksAnimator = new TileAnimator(tlGround, gd.waterrocksLevel1\_ground, gd.waterrocksseq001, true);    tInvulnerable.schedule(new TimerTask() {  public void run() {  if (invulnerable) {  if (spriteKancil.isVisible()) {  spriteKancil.setVisible(false);  } else {  spriteKancil.setVisible(true);  }  } else {  spriteKancil.setVisible(true);  }  }  }, 0, 100);    gd.updateLayerManagerForLevel1(lm);  playerBackgroundSound.setLoopCount(999);  spriteKancil.defineReferencePixel(spriteKancil.getWidth() / 2, spriteKancil.getHeight() / 2);  playerBackgroundSound.start();  t.scheduleAtFixedRate(tlGroundWaterAnimator, 0, gd.waterseq001Delay);  t.scheduleAtFixedRate(spriteKancilAnimator, 0, gd.spriteKancilseq002Delay);  t.scheduleAtFixedRate(tlGroundWaterRocksAnimator, 0, gd.waterrocksseq001Delay);  t.scheduleAtFixedRate(spriteIbuKancilAnimator, 0, gd.spriteIbuKancilseq001Delay);  new Thread(spritesEnemyRandomMovement).start();  new Thread(spritesAnimalRandomMovement).start();  }  public void paint(Graphics g) {  super.paint(g);  lm.paint(g, 0, 0);  g.setFont(Font.getFont(Font.FACE\_MONOSPACE, Font.STYLE\_BOLD, Font.SIZE\_LARGE));  g.setColor(255, 175, 0);  g.drawImage(imageFruit, getWidth() - imageFruit.getWidth(), 0, Graphics.TOP | Graphics.LEFT);  g.drawString(fruits + "", getWidth() - 20, 25, Graphics.TOP | Graphics.LEFT);  for (int j = 5; j < live \* 10; j += 10) {  g.drawImage(imageHeart, j, 5, Graphics.TOP | Graphics.LEFT);  }  }  public void run() {  while (!interrupted) {  input(getKeyStates());  checkGravity(spriteKancil);  checkCollision(spritesEnemy);  checkHealth();  adjutViewPort();  checkBonus();  repaint();  try {  Thread.sleep(20);  } catch (InterruptedException ex) {  ex.printStackTrace();  }  }  }  private void input(int keyStates) {  if ((keyStates & RIGHT\_PRESSED) != 0) {  if (lastDirection != RIGHT) {  lastDirection = RIGHT;  spriteKancil.setTransform(Sprite.TRANS\_NONE);  invulnerable = false;  }  spriteKancilAnimator.setMoving(true);  if (spriteKancil.getX() > tlBackground.getWidth() - spriteKancil.getWidth()) {  spriteKancilAnimator.setMoving(false);  return;  }  spriteKancil.move(3, 0);  } else if ((keyStates & LEFT\_PRESSED) != 0) {  if (lastDirection != LEFT) {  lastDirection = LEFT;  spriteKancil.setTransform(Sprite.TRANS\_MIRROR);  }  spriteKancilAnimator.setMoving(true);  if (spriteKancil.getX() < 0) {  spriteKancilAnimator.setMoving(false);  return;  }  spriteKancil.move(-3, 0);  } else {  spriteKancilAnimator.setMoving(false);  }  if ((keyStates & UP\_PRESSED) != 0 && jump < 15) {  spriteKancil.setFrame(3);  spriteKancil.move(0, -15);  jump++;  }  }  private void checkBonus() {  int col\_bonus = (spriteKancil.getRefPixelX() / 32);  int row\_bonus = ((spriteKancil.getRefPixelY() + 5) / 32);  try {  if (tlBonus.getCell(col\_bonus, row\_bonus) != 0) {  tlBonus.setCell(col\_bonus, row\_bonus, 0);  fruits++;  try {  playerBonusEffect.start();  } catch (MediaException ex) {  }  }  } catch (IndexOutOfBoundsException ioobe) {  }  }  private void checkGravity(Sprite sprite) {  int col\_path = (sprite.getRefPixelX() - sprite.getWidth() / 2 + 15) / 16;  int col\_path2 = (sprite.getRefPixelX() + sprite.getWidth() / 2 - 15) / 16;  int row\_path = (sprite.getRefPixelY() + sprite.getHeight() / 2) / 16;  try {  if (tlPath.getCell(col\_path, row\_path) != 0  || tlPath.getCell(col\_path2, row\_path) != 0  || sprite.collidesWith(spritesAnimal[7], true)  || sprite.collidesWith(spritesAnimal[8], true)  || sprite.collidesWith(spritesAnimal[9], true)) {  for (int i = 7; i < 10; i++) {  if (sprite.collidesWith(spritesAnimal[i], true)) {  if (gd.animalsDirection[i] == LEFT) {  sprite.move(-gd.animalsSpeed[i], 0);  } else {  sprite.move(gd.animalsSpeed[i], 0);  }  }  }  jump = 0;  return;  }  } catch (IndexOutOfBoundsException ioobe) {  }  if (sprite.getY() > getHeight()+sprite.getHeight()) {  live--;  }  sprite.move(0, GRAVITY);  }  private void adjutViewPort() {  int vx = spriteKancil.getX() - spriteKancil.getWidth() / 2;  int vy = spriteKancil.getY();  if (vx <= 0) {  vx = 0;  }  if (vy <= 0) {  vy = 0;  }  if (vy + getHeight() > tlBackground.getHeight()) {  vy = tlBackground.getHeight() - getHeight();  }  if (vx + getWidth() > tlBackground.getWidth()) {  vx = tlBackground.getWidth() - getWidth();  }  this.lm.setViewWindow(vx, vy, getWidth(), getHeight());  }  private void checkCollision(Sprite[] enemies) {  for (int i = 0; i < enemies.length; i++) {  if (spriteKancil.collidesWith(enemies[i], true)) {  if (!invulnerable) {  live--;  }  invulnerable = true;  tInvulnerable.schedule(new TimerTask() {  public void run() {  invulnerable = false;  }  }, 3000);  }  }  if (spriteKancil.collidesWith(spriteIbuKancil, true)) {  midlet.gameOver(fruits);  }  }  private void checkHealth() {  if (live <= 0) {  interrupted = true;  playerBackgroundSound.close();  midlet.gameOver(fruits);  }  }  } |

A.6 GameOverCanvas.java

|  |
| --- |
| import java.io.IOException;  import javax.microedition.lcdui.Canvas;  import javax.microedition.lcdui.Graphics;  import javax.microedition.lcdui.Image;  /\*\*  \*  \* @author egijelek  \*/  public class GameOverCanvas extends Canvas{    private int score;  private Image background;  private MidletSiKancil midlet;  public GameOverCanvas (int score, MidletSiKancil midletSiKancil) throws IOException{  this.score = score;  this.midlet = midletSiKancil;  background = Image.createImage(getClass().getResourceAsStream("/res/GameOver.png"));      }    protected void paint(Graphics g) {  g.drawImage(background, 0, 0, Graphics.TOP | Graphics.LEFT);  g.drawString("Skor = " +score+"", getWidth()/2-15, getHeight()/2, Graphics.TOP | Graphics.LEFT);  }  protected void keyPressed(int keyCode) {  super.keyPressed(keyCode);  midlet.mainMenu();    }} |

A.7 SpriteAnimator.java

|  |
| --- |
| import java.util.TimerTask;  import javax.microedition.lcdui.game.Sprite;  /\*\*  \*  \* @author egijelek  \*/  public class SpriteAnimator extends TimerTask {  private boolean moving = false;  private Sprite sprite;  public SpriteAnimator(Sprite sprite, boolean moving) {  this.sprite = sprite;  this.moving = moving;  }  public void run() {  if (!moving) {  sprite.setFrame(0);  return;  }  sprite.nextFrame();  }  public void setMoving(boolean moving) {  this.moving = moving;  }  } |

A.8 SpritesRandomMovement.java

|  |
| --- |
| import java.util.Timer;  import javax.microedition.lcdui.game.GameCanvas;  import javax.microedition.lcdui.game.Sprite;  import javax.microedition.lcdui.game.TiledLayer;  /\*\*  \*  \* @author egijelek  \*/  public class SpritesRandomMovement implements Runnable {  private static final int BUFFER = 75;  private boolean interrupted;  private int[] speeds;  private final int[] defaultSpeeds;  private int[] delays;  private int[] lastDirections;  private int[] directions;  private Sprite[] sprites;  private Sprite sprite;  private TiledLayer path;  private SpriteAnimator animator;  private Timer t;  private boolean fly;  public SpritesRandomMovement(Sprite[] sprites, int[] speeds, int[] delays, int[] directions, TiledLayer tlPath, Sprite sprite, boolean fly) {  this.fly = fly;  this.sprites = sprites;  this.sprite = sprite;  this.path = tlPath;  this.directions = directions;  this.speeds = speeds;  this.defaultSpeeds = speeds;  this.delays = delays;  lastDirections = new int[sprites.length];  t = new Timer();  for (int i = 0; i < this.sprites.length; i++) {  sprites[i].defineReferencePixel(sprites[i].getWidth() / 2, sprites[i].getHeight() / 2);  t.scheduleAtFixedRate(new SpriteAnimator(sprites[i], true), 0, delays[i]);  }  }  public void run() {  while (!interrupted) {  for (int i = 0; i < sprites.length; i++) {  checkGravity(sprites[i]);  if (directions[i] == GameCanvas.LEFT) {  if (lastDirections[i] != directions[i]) {  lastDirections[i] = directions[i];  sprites[i].setTransform(Sprite.TRANS\_NONE);  }  sprites[i].move(-speeds[i], 0);  if (!fly) {  if (sprites[i].getX() + BUFFER < sprite.getX()  && sprites[i].getX() + BUFFER + 30 > sprite.getX()  && sprites[i].getY() - BUFFER < sprite.getY()) {  directions[i] = GameCanvas.RIGHT;  }  }  if (checkEdge(sprites[i])) {  directions[i] = GameCanvas.RIGHT;  }  } else if (directions[i] == GameCanvas.RIGHT) {  if (lastDirections[i] != directions[i]) {  lastDirections[i] = directions[i];  sprites[i].setTransform(Sprite.TRANS\_MIRROR);  }  sprites[i].move(speeds[i], 0);  if (!fly) {  if (sprites[i].getX() - BUFFER > sprite.getX()  && sprites[i].getX() - BUFFER - 30 < sprite.getX()  && sprites[i].getY() - BUFFER < sprite.getY()) {  directions[i] = GameCanvas.LEFT;  }  }  if (checkEdge(sprites[i])) {  directions[i] = GameCanvas.LEFT;  }  }  }  try {  Thread.sleep(20);  } catch (InterruptedException ex) {  ex.printStackTrace();  }  }  }  public void checkGravity(Sprite sprite) {  if (!fly) {  int col = sprite.getRefPixelX() / path.getCellWidth();  int row = (sprite.getRefPixelY() + sprite.getHeight() / 2) / path.getCellHeight();  if (path.getCell(col, row) != 0) {  return;  }  sprite.move(0, 8);  }  }  public boolean checkEdge(Sprite sprite) {  if (!fly) {  int col\_a = (sprite.getRefPixelX() - (sprite.getWidth() / 2)) / path.getCellWidth();  int col\_b = (sprite.getRefPixelX() + (sprite.getWidth() / 2)) / path.getCellWidth();  int row = (sprite.getRefPixelY() + sprite.getHeight() / 2) / path.getCellHeight();  if (path.getCell(col\_a, row) == 0 || path.getCell(col\_b, row) == 0) {  return true;  } else {  return false;  }  } else {  if (sprite.getX() <= 0 || sprite.getX() >= path.getWidth() || sprite.collidesWith(path, false)) {  return true;  } else {  return false;  }  }  }  } |

A.9 TileAnimator.java

|  |
| --- |
| import java.util.TimerTask;  import javax.microedition.lcdui.game.TiledLayer;  /\*\*  \*  \* @author egisolehhasdi  \*/  public class TileAnimator extends TimerTask {  private boolean moving = true;  private boolean forward = true;  private TiledLayer tiledLayer;  private int animatedTileIndex;  private int[] sequence;  private int sequenceIndex;  public TileAnimator(TiledLayer tiledLayer, int animatedTileIndex, int[] sequence, boolean forward) {  this.tiledLayer = tiledLayer;  this.animatedTileIndex = animatedTileIndex;  this.sequence = sequence;  this.forward = forward;  }  public void run() {  if (!this.moving) {  return;  }  if (forward) {  if (++this.sequenceIndex >= this.sequence.length) {  sequenceIndex = 0;  }  } else {  if (--this.sequenceIndex < 0) {  sequenceIndex = this.sequence.length - 1;  }  }  this.tiledLayer.setAnimatedTile(this.animatedTileIndex, this.sequence[sequenceIndex]);  }  public void forward() {  this.forward = true;  this.moving = true;  }  public void backward() {  this.forward = false;  this.moving = true;  }  public void setMoving(boolean isMoving) {  this.moving = isMoving;  }  } |

LAMPIRAN B  
BIODATA PENULIS



Nama : Egi Soleh Hasdi

NIS : 0910113314

Kelas : XII-M

Jurusan : Rekayasa Perangkat Lunak

Sekolah : SMK Negeri 4 Bandung

Tempat Lahir : Bandung

Tanggal Lahir : 16 Juni 1993

Alamat : Jl. H.Nawawi no. 13/107 RT 05 / RW 04

Kec. Lengkong Kel. Burangrang Bandung

Telepon : 087825441181

Email : egienfinity@gmail.com