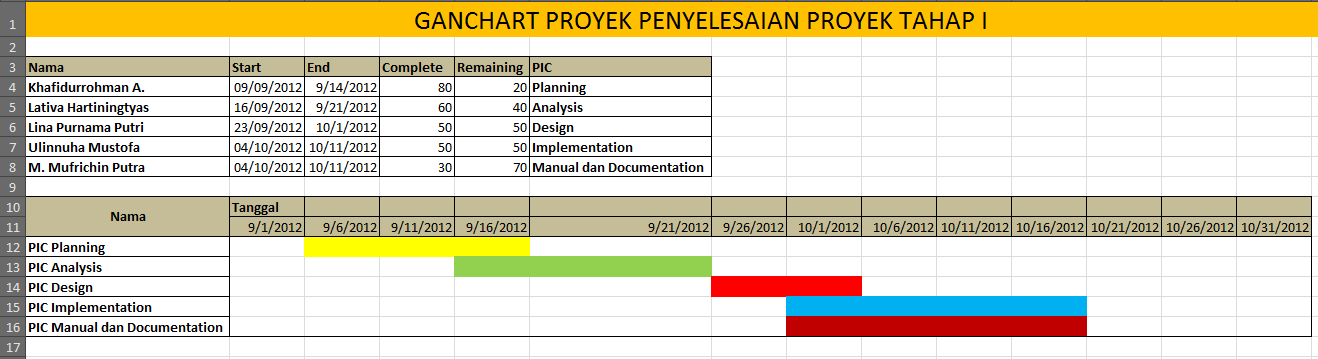
**PROJECT PLAN**

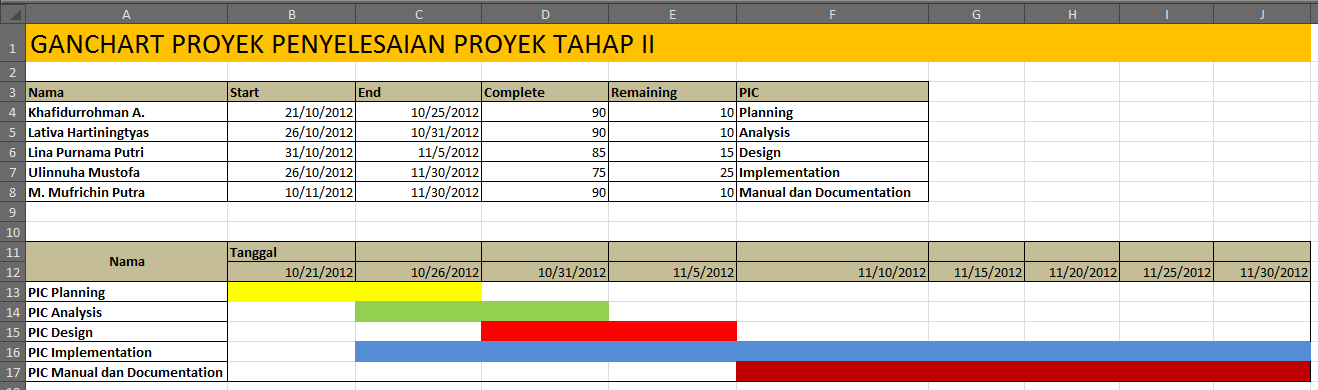
1. **Nama Anggota:**
2. Khafidurrohman Agustianto (100533402595)
3. Lina Purnama Putri (100533402605)
4. M. Ulinnuha Musthofa (100533402612)
5. Lativa Hartiningtyas (100533402630)
6. Muhammad Mufrichin Putra (100533404330)
7. **Apa yang Akan Dibuat:**

Yang akan dibuat pada *project* ini adalah aplikasi KBlog, yaitu suatu aplikasi yang menghitung tanggal periode dan siklus menstruasi tanpa harus mengingat-ingat atau mencatat secara manual. Pada program ini juga terdapat juga menerima masukan dari pengguna (*user*) berupa data pribadi yang menyangkut tanggal menstruasi, golongan darah suami / istri, tanggal lahir suami / istri, dan siklus menstruasi. Pada aplikasi ini juga terdapat *Ovulation and Fertility Forecast*, yaitu untuk memperkirakan tingkat kesuburan dan saat-saat subur dari seorang istri. Pada program ini terdapat *Password Protection*, yang mana ini untuk menjaga keamanan dan hanya user yang berkepentingan saja yang bisa membuka dan melihat aplikasi ini.

Notification berisi hal-hal seputar Menstruasi, Pil dan alat Kontrasepsi, Pil Multivitamin, dan lain sebagainya.

Pada aplikasi ini juga terdapat Note untuk memberi atau menuliskan catatan yang dimasukkan oleh user yang berhubungan dengan hal-hal siklus menstruasi user.

1. **Jadwal Kegiatan**



**Penyelesaian Proyek tahap I**

Planning : 09 September 2012 – 14 September 2012

Analysis : 16 September 2012 – 21 September 2012

Design : 23 September 2012 – 01 Oktober 2012

Implementation : 04 Oktober 2012 – 11 Nopember 2012

Manual dan Documentation : 04 Oktober 2012 – 10 Nopember 2012

**Penyelesaian Proyek Tahap II**

Planning : 21 Oktober 2012 – 25 Oktober 2012

Analysis : 26 Oktober 2012 – 31 Oktober 2012

Design : 31 Oktober 2012 – 5 Nopember 2012

Implementation : 26 Oktober 2012 – 30 Nopember 2012

Manual dan Documentation : 10 Nopember 2012 – 30 Nopember 2012

1. **Software Development Method**

Pengembangan perangkat lunak (PL) dapat dianggap sebagai lingkaran pemecahan masalah. Untuk menyelesaikan masalah besar, dipecah menjadi kecil terus-menerus sampai paling kecil, kemudian diselesaikan (*recursive*). Pada pertengahan tahun 60 sampai 70-an banyak dikembangkan sistem-sistem

perangkat lunak yang besar. Sistem-sistem yang dikembangkan ini banyak yang dipandang tidak efisien, kurang berhasil, bahkan banyak yang gagal. Kegagalan ini disebabkan karena tidak tersedianya teknik pengembangan perangkat lunak yang baik. Pada awal tahun 70-an mulai muncul metodologi-metodologi pengembangan perangkat lunak yang cukup baik.

Pengembangan perangkat lunak dapat diartikan sebagai proses membuat suatu perangkat lunak baru untuk menggantikan perangkat lunak lama secara keseluruhan atau memperbaiki perangkat lunak yang telah ada. Agar lebih cepat dan tepat dalam mendeskripsikan solusi dan mengembangkan perangkat lunak, juga hasilnya mudah dikembangkan dan dipelihara, maka pengembangan perangkat lunak memerlukan suatu metodologi khusus. Metodologi pengembangan perangkat lunak adalah suatu proses pengorganisasian kumpulan metode dan konvensi notasi yang telah didefinisikan untuk mengembangkan perangkat lunak. Secara prinsip bertujuan untuk membantu menghasilkan perangkat lunak yang berkualitas. Penggunaan suatu metodologi sesuai dengan persoalan yang akan dipecahkan dan memenuhi kebutuhan pengguna akan menghasilkan suatu produk perekayasaan yang berkualitas dan terpelihara serta dapat menghindari masalah-masalah yang sering terjadi seperti estimasi penjadwalan dan biaya, perangkat lunak yang tidak sesuai dengan keinginan pengguna dan sebagainya.

Metodologi pengembangan perangkat lunak (atau disebut juga model proses atau paradigma rekayasa perangkat lunak) adalah suatu strategi pengembangan yang

memadukan proses, metode, dan perangkat (tools). Menurut Pressman (1997) Komponen metodologi pengembangan perangkat lunak dapat

dibagi dalam tiga unit, yaitu :

* Metode, yaitu suatu cara atau teknik pendekatan yang sistematik yang dipergunakan untuk mengembangkan perangkat lunak. Metode ini mencakup : Perencanaan proyek dan perkiraan, analisis keperluan sistem dan perangkat lunak, perancangan struktur data, arsitektur program, prosedur algoritma, Coding, uji coba dan pemeliharaan.
* Alat bantu (Tools), yaitu alat-alat (manual atau otomatis) yang mendukung pengembangan perangkat lunak. Terdapat 2 alat Bantu yang dapat digunakan yaitu: alat Bantu manual dan alat Bantu otomatis.
* Prosedur, yang dipergunakan untuk mendefinisikan urut-urutan pekerjaan (daur) dari metode dan alat bantu tersebut.

Pada program ini Software Development Method-nya adalah dengan menggunakan pengembangan software The V-Model, yaitu suatu merupakan proses pengembangan perangkat lunak (juga berlaku untuk pengembangan hardware) yang dapat dianggap sebagai perluasan dari model air terjun . Alih-alih bergerak turun dengan cara yang linear, langkah-langkah proses yang bengkok ke atas setelah pengkodean fase, untuk membentuk bentuk V yang khas. The V-Model menunjukkan hubungan antara setiap fase siklus hidup pengembangan dan fase terkait dari pengujian . Sumbu horisontal dan vertikal merupakan kelengkapan waktu atau proyek (kiri ke kanan) dan tingkat abstraksi (yang kasar-butiran menonjol abstraksi), masing-masing.

Tahapan-tahapan pada V-Models adalah sebagai berikut :

* Unit pengujian

Dalam pemrograman komputer, unit testing adalah metode dimana unit individu dari kode sumber yang diuji untuk menentukan apakah mereka cocok untuk digunakan. Unit adalah bagian terkecil dari diuji aplikasi. Dalam pemrograman prosedural unit mungkin merupakan fungsi individual atau prosedur. Unit test yang dibuat oleh programmer atau kadang-kadang oleh penguji kotak putih. Tujuannya adalah untuk memverifikasi kode logika internal dengan menguji setiap cabang mungkin dalam fungsi, juga dikenal sebagai cakupan tes. Alat analisis statis digunakan untuk memudahkan dalam proses ini, di mana variasi data masukan yang dilewatkan ke fungsi untuk menguji setiap kemungkinan kasus eksekusi.

* Integrasi pengujian

Dalam pengujian integrasi modul terpisah akan diuji bersama-sama untuk mengekspos kesalahan dalam antarmuka dan dalam interaksi antara komponen terintegrasi. Pengujian biasanya kotak hitam sebagai kode ini tidak langsung diperiksa untuk kesalahan.

* Pengujian Sistem

Pengujian sistem akan membandingkan spesifikasi sistem terhadap sistem yang sebenarnya. Setelah tes integrasi selesai, tingkat tes berikutnya adalah tes sistem. Pengujian sistem memeriksa apakah produk terintegrasi memenuhi persyaratan yang ditentukan. Mengapa ini masih diperlukan setelah tes komponen dan integrasi? Alasan untuk ini adalah sebagai berikut:

* Alasan Pengujian Sistem

Pada tingkat yang lebih rendah pengujian, pengujian dilakukan terhadap spesifikasi teknis, yaitu, dari perspektif teknis dari produsen software. Tes sistem, meskipun, melihat sistem dari perspektif pelanggan dan pengguna di masa depan. Para penguji memvalidasi apakah persyaratan benar-benar dan tepat bertemu.

Contoh: Pelanggan (yang telah dipesan dan dibayar untuk sistem) dan pengguna (yang menggunakan sistem) bisa menjadi berbagai kelompok orang atau organisasi dengan kepentingan mereka sendiri yang spesifik dan persyaratan sistem.

Banyak fungsi dan sistem hasil karakteristik dari interaksi semua komponen sistem, akibatnya, mereka hanya terlihat pada tingkat seluruh sistem dan hanya dapat diamati dan diuji di sana.

* Pengguna Pengujian Penerimaan

Pengujian penerimaan adalah tahap pengujian digunakan untuk menentukan apakah sistem memenuhi persyaratan yang ditentukan dalam tahap analisis persyaratan. Desain tes penerimaan berasal dari dokumen persyaratan. Tahap uji penerimaan adalah fase yang digunakan oleh pelanggan untuk menentukan apakah akan menerima sistem atau tidak.

* Pengujian Penerimaan Membantu

Untuk menentukan apakah sistem memenuhi kriteria penerimaan atau tidak.

untuk memungkinkan pelanggan untuk menentukan apakah akan menerima sistem atau tidak.

untuk menguji perangkat lunak di "dunia nyata" oleh audiens.

Tujuan pengujian penerimaan:

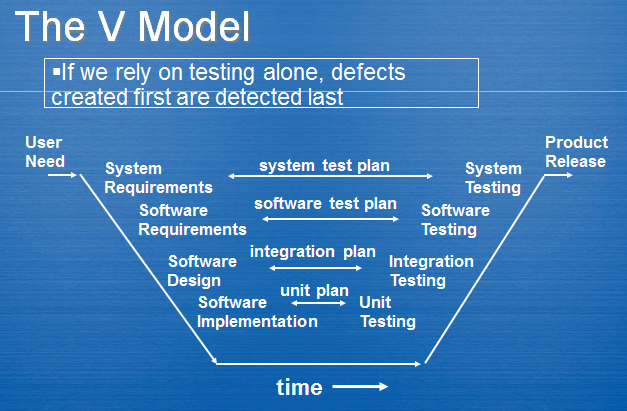
untuk memverifikasi sistem atau perubahan sesuai dengan kebutuhan aslinya.

Prosedur kriteria penerimaan:

* Fungsi persyaratan.
* Persyaratan kinerja.
* Antarmuka persyaratan mutu.
* Kualitas perangkat lunak persyaratan keseluruhan.
* Mengembangkan rencana penerimaan:
* Uraian proyek.
* Pengguna tanggung jawab.
* Penerimaan deskripsi.

Pengujian Rilis

Pengujian rilis adalah fase yang menentukan apakah software ini cocok untuk organisasi pengguna akhir. Bagaimana kompatibilitas dengan sistem lain meyakinkan, Apakah kinerja perangkat lunak dioptimalkan.



System Requirements → menggunakan spesifikasi OS Mobile Android versi Eclare ke atas (Frozen Yogurt, Gingerbread, Ice Cream Sandwicth, Jellybean)

Software Requirements → Sotware ini mampu menerima masukan dari user berupa data pribadi menyangkut tanggal menstruasi, golongan darah suami/istri, rhesus darah suami/istri, tanggal lahir suami/istri, dan siklus menstruasi

* Menstrual Cycle and Period
* Ovulation and Fertility Forecast
* Password Protection
* Notification (Menstruation, Contraceptive Pil, Multivitamin Pil)
* Note, dll

Software Design → Design dari aplikasi menggunakan Adobe Photoshop CS5, tampilan awal berupa menu Calender, About, Setting. Pada tampilan home ini terdapat About dan Help. Pada menu selanjutnya terdapat Calender, dan terdapat Suggestion yang berisi tentang tips-tips kehamilan. pada menu Setting terdapat penjelasan mengenai software, versi-versi software, developer software, dan fitur-fitur yang ada pada software ini, dan terdapat keterangan mengenai menstruasi, kehamilan, masa subur, dan ovulasi. Pada menu Setting terdapat pengaturan username, interval, dan masa (menstruasi).

Software Implementation → menggunakan Adobe Photoshop, Corel Draw, PhoneGap Framework, jQTouch, Java Runtime Environment, Java Development Kit, Eclipse, HTML 5, Notepad++, Browser (google chrome, mozilla firefox) untuk mengembangkan software ini.