



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODE PENELITIAN

3.1 Objek Penelitian

Pada penyusunan skripsi ini penulis melakukan penelitian tentang pengembangan aplikasi berbasis *android* untuk membantu pemilik tempat kursus maupun bimbingan belajar menyusun jadwal untuk pengajar dan siswanya. Untuk melakukan analisis dan pengumpulan data, penulis melakukan observasi pada aplikasi *scheduling* yang sudah beredar di *Internet* yang akan digunakan sebagai referensi dalam membuat aplikasi penjadwalan.

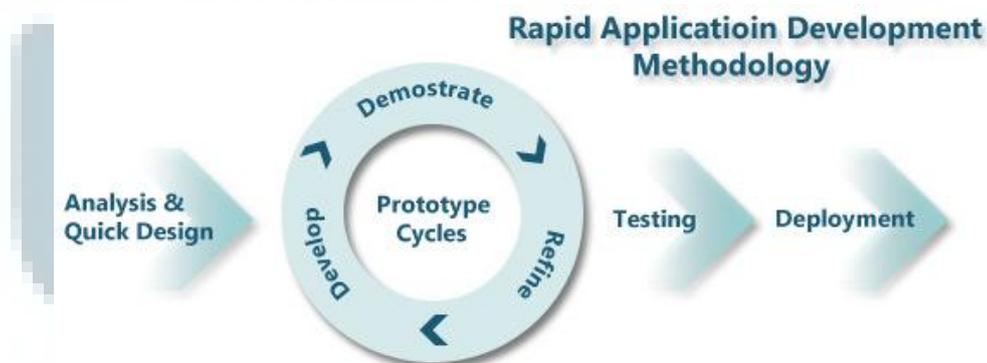
3.2 Metode Penelitian

Dalam merancang aplikasi penulis akan menggunakan *System Development Life Cycle* (SDLC). SDLC merupakan model yang digunakan untuk menerangkan tahapan-tahapan yang terlibat dalam proses membangun sebuah sistem informasi. SDLC memiliki beberapa tipe model dan penulis memilih menggunakan model *Rapid Application Development* (RAD) karena memiliki keunggulan dibandingkan dengan model *Waterfall* seperti yang digambarkan pada tabel 3.1.

Tabel 3.1 Perbandingan model *Waterfall* dan RAD

<i>Waterfall</i>	<i>Rapid Application Development</i>
Bisa diimplementasikan untuk tipe proyek apa saja	Diimplementasikan dan gampang dikelola untuk proyek skala kecil
<i>System requirement</i> tidak bisa diubah pada saat <i>development</i> sehingga tidak cocok untuk proyek yang kemungkinan <i>requirementnya</i> berubah ditengah-tengah <i>development</i> .	<i>System requirement</i> bisa diubah kapan saja
Interaksi dengan <i>user</i> hanya terjadi pada saat mengumpulkan <i>requirement</i> dan <i>testing</i>	Interaksi dengan <i>user</i> terjadi pada setiap <i>phase development</i>
Tidak ada <i>prototyping</i>	<i>Developer</i> berfokus mendesain <i>prototype</i> dan mengembangkan <i>prototype</i> tersebut menjadi produk jadi.
<i>Software</i> hanya <i>ditest</i> pada tahap akhir <i>development</i> , jika ada <i>bugs</i> pada <i>software</i> pada tahap awal pengembangan tapi baru ditemukan pada tahap akhir maka akan mempengaruhi seluruh <i>code software</i> .	Testing dilakukan terus-menerus pada tahap <i>prototyping</i> .
Tidak ada <i>software</i> yang diproduksi sampai tahap akhir siklus pengembangan	<i>Software</i> diproduksi pada tahap awal pengembangan

Rapid Application Development yang memiliki tahapan sebagai berikut:



Gambar 3.1 Metode *Rapid Application Development*

3.2.1 Analisis & Quick Design

Pada tahap ini penulis mengumpulkan *software requirement specification* dan untuk membuat rancangan aplikasi dan menentukan bagaimana alur proses aplikasi ini akan berjalan. Dalam penelitian ini pengumpulan data dilakukan dengan dua cara, yaitu studi literatur dan observasi:

1. Studi Pustaka

Pada tahap studi pustaka penulis mengumpulkan data dari berbagai macam sumber seperti buku, jurnal, dan artikel yang akan digunakan sebagai referensi dalam melakukan penelitian.

2. Observasi

Observasi dilakukan dengan cara mengumpulkan data dan informasi yang dibutuhkan dalam perancangan dari aplikasi penjadwalan yang sudah ada.

Setelah mengetahui *requirement* yang dibutuhkan penulis membuat diagram yang akan menjadi rancangan aplikasi yang akan dibuat berupa *Flowchart*, *Data Flow Diagram* dan *Entity Relationship Diagram*.

3.2.2 Prototype Cycles

Pada tahap ini penulis membangun aplikasi berdasarkan siklus *development*, *demonstrate*, dan *refine* sampai aplikasi selesai dibuat. Penulis membuat *design database* dan *design website*. Selanjutnya penulis membuat *prototype* aplikasi berdasarkan *design* tersebut. Lalu *prototype* aplikasi di *review*

dan apabila masih ada fungsi yang kurang pada aplikasi maka penulis akan kembali pada tahap *develop* sampai aplikasi selesai dibuat.

3.2.3 Testing

Pada tahap *testing* penulis akan mencoba menggunakan aplikasi yang sudah dibuat dalam *prototype cycle* untuk mengetahui apakah ada *bug* atau fitur yang kurang dalam aplikasi. Jika terdapat kesalahan dalam aplikasi maka penulis akan kembali ke tahap *prototype cycles* untuk memperbaiki atau menambah fitur aplikasi yang penulis buat.

3.2.4 Deployment

Seperti yang telah disampaikan pada pembatasan masalah, penulis tidak melakukan penelitian sampai tahap ini.

3.2.5 Rencana Kegiatan

Dalam membuat aplikasi penjadwalan yang akan berlangsung selama 3 bulan penulis menggunakan *timeline* sebagai berikut:

U M N



Gambar 3.2 Gantt Chart Rencana Kegiatan

3.3 Spesifikasi Teknis Penelitian

Dalam melakukan penelitian ini penulis menggunakan komputer dengan *processor* AMD FX 8320 3.5Ghz, *random access memory* sebesar 8 GB, kartu grafis AMD 7850 4 GB, *hard disk* sebesar 1 TB, 2 buah monitor 19 inch, *mouse* dan *keyboard* untuk *input device* serta Samsung Note 3 untuk *testing* aplikasi.

Untuk perangkat lunak penulis menggunakan *integrated development environment* NetBeans 8.0 untuk *programming*, Microsoft Word 2013 untuk menyusun laporan penelitian, serta XAMPP untuk mengetes aplikasi yang dibuat.

