

## BAB II

### TINJAUAN PUSTAKA

#### **2.1 Agile Develepemt Method**

*Agile developmet method* adalah pengembangan perangkat lunak yang mempunyai rentang waktu pendek serta bisa beradaptasi dengan adanya perubahan yang mungkin akan terjadi selama proses dari pembangunan sistem tersebut berlangsung[9]. Salah satu metode yang bisa dikatakan sebagai *agile development method* apabila dia bersifat *iterative, incremental, flexibility, responding to change*, dan *lightweight*. Ada beberapa hal yang penting dari *Agile development method* tersebut adalah pendeskripsian dari sistem yang bisa dimengerti dengan mudah dan dapat digunakan oleh pihak *client*. Mengenai penjelasan yang dimaksud *agile development method* diatas yakni:

a. *Iterative*

*Iterative* ini bisa diimplementasikan dalam proses pembangunan sebagai penyelesaian perangkat lunak dengan cara pendekatan secara periodik guna menemukan solusi dari permasalahan yang diperoleh selama proses pembangunan sistem[10].

b. *Incremental*

*Incremental* ini pada pengembangan perangkat lunak tentu kebutuhan fungsionalitas pada sistem dibagi menjadi beberapa bagian yang lebih kecil. Pada setiap pengembanagn fungsionalitas pada perangkat lunak akan semakin bertambah sehingga akan menjadi suatu perangkat lunak yang utuh[11].

c. *Flexibility*

*Flexibility* ini apabila sebuah perangkat lunak dikembangkan, sistem tersebut bisa beradaptasi dengan kondisi tersebut dikarenakan pengembang sudah melibatkan pihak *client* secara langsung terhadap proses pembangunan perangkat lunak. Oleh karena itu, pengembang harus siap ketika adanya perubahan yang akan terjadi[10].

d. *Responding to change*

*Responding to chage* ini memperbolehkan perubahan terkait *requirement, code, and design* dari perangkat lunak yang sudah dibuat[12].

e. *Lightweight*

*Lightweight* ini pengembang dapat meminimalkan dokumentasi *requirement* dan segmen-segmen yang tidak perlu selama pembuatan perangkat lunak sedang berlangsung dikarenakan situasi ini dapat memperlambat kecepatan dan waktu dalam pembuatan perangkat lunak[11].

## **2.2 Extreme Programming**

Menurut Kent beck, *Extreme Programming* (XP) merupakan metode yang ringan pada tim yang kecil dan menengah yang sedang mengembangkan suatu sistem atau perangkat lunak sesuai dengan *requirement* yang masih belum jelas atau dapat berubah dengan begitu cepat.

Pengembang sistem yang menggunakan metode *Extreme Programming* (XP) ini akan melakukan tahap awal yakni *stories* oleh *customer* sehingga dapat mendeskripsikan kebutuhan fungsionalitas pada sistem yang akan dibuat. *Stories* yang sudah didapat tadi merupakan fungsionalitas yang kecil dan membutuhkan waktu kira-kira satu sampai 2 minggu yang nantinya akan diuji dan dikodekan. Para pembuat sistem ini akan memberikan informasi estimasi waktu pada setiap *user story*, dan *customer* dapat menentukan nilai dan biaya, dan *story* manakah yang akan dikerjakan lebih awal. Pengembang akan melakukan *iterative* secara berjenjang dan pengembang juga akan menyampaikan hasil dari pengerjaan pada *customer* setiap 2 minggu. Kemudian setelah itu *customer* dapat memilih *story* yang akan dikerjakan untuk minggu depannya[13].

## **2.3 Personal Extreme Programming**

Metode *Personal Extreme Programming* (PXP) merupakan pengembangan sebutan bagi metode *Extreme Programming* (XP) yang masih memiliki banyak kesamaan diantara keduanya namun perbedaan sebutan tersebut dikarenakan metode PXP hanya dikembangkan oleh satu orang saja, sementara pada metode XP dapat dikembangkan oleh lebih dari dua orang pengembang. Meskipun secara garis besar kedua metode ini sama namun tetap saja metode ini tidak dapat digunakan untuk pengembangan sistem berskala besar [14].

Walaupun metode PXP termasuk dalam pengembangan perangkat lunak secara personal namun harus memiliki *iterative*, fleksibel, dan responsif terhadap pengembangan perangkat lunak seperti halnya dengan metode *Extreme Programming* atau bisa disebut

dengan XP, metode tidak dapat digunakan untuk pengembangan perangkat lunak berskala besar[15].

Metode PXP ini awal mulanya diterapkan pada tahun 2009 oleh Dzhyrov dengan penelitian yang bertema “*Personal Extreme Programming An Agile Process For Autonomonus Development*”[16]. Pada keadaan ini metode *Extreme Programming* tidak bisa dikatakan sama ataupun mirip secara keseluruhan dengan *Personal Extreme Programming* dikarenakan pada tahap implementasi kedua metode ini memiliki 6 hal yang bisa diterapkan pada metode PXP dan XP yakni :

1. *Continous Integration*
2. *Simple Design*
3. *Small Release*
4. *Refactoring*
5. *Test Driven Development*
6. *Spike Solution*

Sedangkan pada tahap pelaksaan yang lain yang terdapat pada *Extreme programming* apabila ingin diterapkan pada metode *Personal Extreme Programming* harus dimodifikasi terlebih dahulu agar pada tahap pelaksanaan tersebut bisa berjalan pada metode PXP[17].

## **2.4 User Story**

Rumusan berupa kebutuhan-kebutuhan paling mendasar dari sistem yang diinginkan oleh pengguna merupakan *user story*. Rumusan dari pengguna tersebut yang akan menjadi acuan bagi pengembang sistem untuk menyusun dan mengimplementasikan arsitektur sistem yang akan dibangun dengan menyusun *prototype* sistem secara sederhana. Waktu pengembangan sistem yang telah dirancang dapat diestimasikan berdasarkan seberapa banyak jumlah *user story* yang berhasil didokumentasikan oleh pengguna. *User story* digunakan sebagai pedoman pengimplementasian perangkat lunak yang akan dibangun[18]. *User story* ini bermanfaat untuk alat komunikasi antara pengembang dan klien, dimana pihak yang terkait akan lebih mudah dalam memahami apa saja kebutuhan yang dibutuhkan oleh klien sehingga dapat di kerjakan oleh pengembang. Developer dan klien akan berdiskusi mengenai *user story* yang telah dibangun selama proses pengembangan aplikasi. Motif dalam pembuatan *user story* ini berkesinambungan dengan dimensi *WHO*, *WHAT*, dan *WHY*, akan tetapi ada beberapa rancangan juga

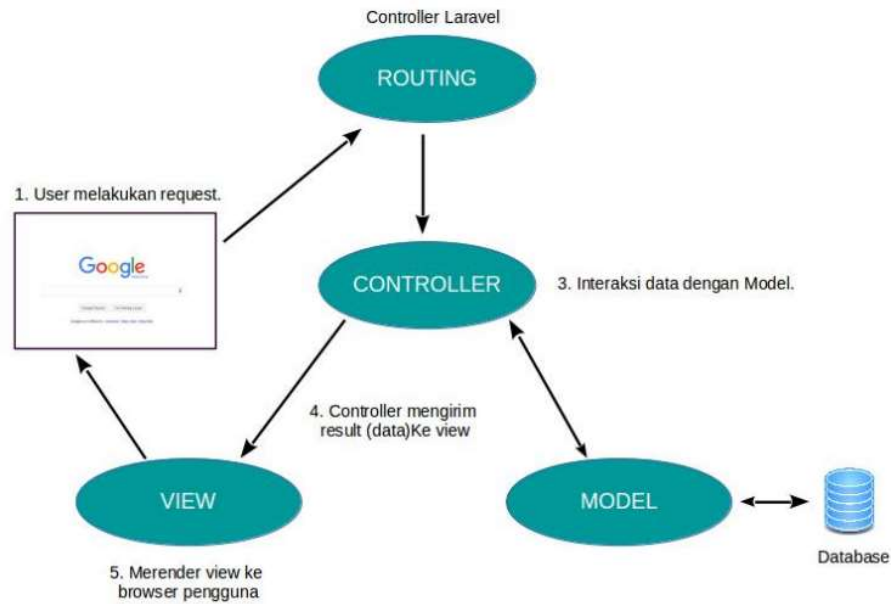
bisa didapat dalam literature atau website untuk mewakili dimensi ini (misalnya, MikeCohn's selaku < tipe pengguna>, aku ingin < beberapa tujuan> sehingga < beberapa alasan>)[18]. Kemudian untuk penulisan *user stories* terdapat karakteristik dalam menuliskan sebuah *story* yang baik harus memenuhi kategor berikut *Independent*, *Negotiable*, *Valuble to purchaser or users*, *Estimable*, *Small*, dan *Testable*[19].

## 2.5 Laravel

Laravel merupakan sebuah *framework open source* yang terbentuk dengan gabungan dari banyak fitur terbaik *framework CodeIgniter*, *Yii*, *ASP.NET MVC*, *Ruby on Rails*, *Sinatra* dan lain-lain. *Framework* ini dinilai lebih unggul dari *framework* lainnya karena memiliki performa yang lebih cepat, *reload* data yang lebih stabil, keamanan data yang lebih baik, memiliki *library* yang sudah siap digunakan dan fitur pengelolaan migrasi untuk membuat skema tabel pada *database*[20].

## 2.6 Model View Controller

MVC yang merupakan singkatan dari *Model*, *View* dan *Controller* adalah konsep pendekatan bagi *software* dalam memisahkan komponen-komponen aplikasi seperti manipulasi data, *controller* dan *user interface*. Pengembangan sistem informasi berbasis *website* yang dibangun menggunakan *framework Laravel* membutuhkan konsep alur kerja MVC.



**Gambar 2. 1** Alur kerja MVC pada framework Laravel

Arsitektur pada *framework Laravel* dengan masing-masing fungsinya terdiri dari lima konsep yaitu *routes* yang berfungsi sebagai gerbang akses untuk setiap *request* dari alur yang telah ditentukan, *controller* yang berfungsi sebagai jembatan antara *model* dan *view*, *model* berfungsi untuk mengelola tabel-tabel yang ada pada *database*, *view* yang berfungsi untuk menampilkan data ke dalam *browser* agar dapat dilihat pengguna serta *migration* yang berfungsi sebagai penyedia sistem kontrol pada skema *database* [21].

## 2.7 Class Responsibilities Collaboration

*Class Responsibilities Collaboration* (CRC) merupakan sebuah kartu indeks yang bisa berguna dalam penunjuk interaksi antar objek. Pada proses pembangunan *Class Responsibilities Collaboration*, dipilah menjadi tiga sub class, *responsibilities*, dan *collaborator*. Kartu dalam perangkat lunak digunakan dalam menentukan class dan apa saja kolaborasi yang akan dibuat[22].

## 2.8 Datatable Serverside Processing

Datatable merupakan *plugin JQuery* yang mempunyai sifat *open source*. Datatable ini mengkonversi pada tampilan *table* html format dengan tampilan dari datatable dengan antarmuka yang intuitif. *Plugin* datatable ini memiliki fitur yang penting untuk mendukung penyajian data. Fitur tersebut ini meliputi pencarian data menggunakan *keyword*, *paginasi*, *sorting*, dan masih banyak lagi fitur yang lainnya

yang dapat digunakan. Proses penyajian data yang dilakukan adalah dengan datatable dapat menggunakan teknik pemrosesan dari *client* (*client-side processing*) maupun pemrosesan dari sisi *server* (*server-side processing*). Akan tetapi pemrosesan kali ini yakni menggunakan *server* (*server-side processing*) sehingga beban *client* untuk menampilkan halaman itu tidak terlalu berat[2].

## 2.9 Notifikasi Email

Email adalah kependekan dari *electronic mail*. *Email* merupakan tempat untuk mengirim surat melalui jalur internet. *E-mail* sendiri ialah salah satu proses pengiriman surat melalui jalur internet dengan waktu yang sangat cepat[23]. Notifikasi email ini akan mengirimkan informasi sesuai apa saja yang dipinjam melalui email.

## 2.10 Penelitian Terkait

Penelitian yang telah dikerjakan tidak lepas dari oleh beberapa penelitian terdahulu yang terkait untuk menunjang keberhasilan penelitian yang akan dikerjakan. Penelitian terkait yang mendukung penelitian ini disebutkan pada tabel 3.0 di bawah ini.

**Tabel 2. 1** *Tabel Relevansi*

No	Nama	Judul	Hasil
1	L. Rusdiana, 2018.	Extreme Programming untuk rancang bangun aplikasi pengelolaan surat keterangan kependudukan	Metode Extreme Programming dapat diterapkan dalam membangun aplikasi yang sederhana tanpa membutuhkan tahapan berulang dari perangkat lunak. Metode XP ini juga sudah membangun aplikasi pengelolaan surat pada Kantor Lurah Rantau Pulut.
2	N. Rochmawati and E. R.	Perancangan sistem informasi inventaris	Banyaknya barang barang penunjang kegiatan dan

	Saputra,2016.	untuk peminjaman dan pengembalian barang di laboratorium jurusan teknik informatika ft unesa	penelitian didalam laboratorium tersebut. Sehingga perlu dibuatkan sistem pencatatan peminjaman laboratorium sehingga barang barang yang dipinjam dapat terinventaris dengan baik
3	U. P. Indonesia, P. Model, and A. Android,2017.	Aplikasi Sistem Informasi Manajemen Sekolah Dasar Pilar Bangsa Untuk Meningkatkan Layanan Pendidikan	Pengembangan sistem informasi manajemen sekolah ini merupakan salah satu bentuk untuk meningkatkan pendidikan. Oleh karena itu dibutuhkanlah pengembangan sistem informasi seperti ini agar dapat meningkatkan efisiensi dan layanan dari berbagai pihak terutama adalah orang tua siswa.