**Studi Kasus : E-Diary**

E-Diary merupakan sistem buku harian elektronik yang nantinya dapat digunakan untuk mencatat kegiatan harian penggunanya dan menyimpannya secara elektronik. Catatan harian yang telah disimpan, dapat dibuka dan dibaca kembali oleh pengguna. Selain itu, pengguna dapat mencari catatan yang telah disimpan menggunakan kata kunci. Karena konsepnya adalah diary, maka catatan yang telah disimpan tidak dapat dihapus maupun diedit dan dalam catatan tersebut tersimpan tanggal dan tempat catatan tersebut ditulis.

Lakukanlah analisis dan desain untuk sistem E-Diary tersebut !

Jawab :

**Requirement Elicitation**

1. Penentuan Aktor

Aktor : pengguna E-Diary

1. Penyusunan Skenario

Skenario yang memungkinkan untuk dibuat adalah : mencatat kegiatan harian, menyimpan kegiatan harian, membaca kegiatan harian, mencari catatan harian.

|  |  |
| --- | --- |
| **Aktor** | Pengguna E-Diary |
| **Nama Skenario** | Menulis catatan harian |
| **Flow Of Events / Urutan Kejadian** | 1. Pengguna E-Diary membuka E-Diary 2. Pengguna E-Diary menuliskan tempat dimana pengguna berada 3. Pengguna menuliskan catatan harian |
| **Alternate Scenario** | - |
| **Exceptional Scenario** | - |

|  |  |
| --- | --- |
| **Aktor** | Pengguna E-Diary |
| **Nama Skenario** | Menyimpan catatan harian |
| **Flow Of Events / Urutan Kejadian** | 1. Pengguna E-Diary selesai menuliskan catatan harian 2. Pengguna E-Diary menyimpan catatan harian 3. E-Diary menyimpan catatan harian pengguna |
| **Alternate Scenario** | - |
| **Exceptional Scenario** | * 1. Pengguna tidak mengisikan catatan harian   2. Sistem tidak menyimpan catatan harian |

|  |  |
| --- | --- |
| **Aktor** | Pengguna E-Diary |
| **Nama Skenario** | Membaca catatan harian |
| **Flow Of Events / Urutan Kejadian** | 1. E-Diary menampilkan daftar catatan harian yang ada. 2. Pengguna E-Diary memilih catatan harian. 3. Pengguna E-Diary membuka catatan harian. 4. Pengguna E-Diary membaca catatan harian. |
| **Alternate Scenario** | - |
| **Exceptional Scenario** | - |

|  |  |
| --- | --- |
| **Aktor** | Pengguna E-Diary |
| **Nama Skenario** | Mencari catatan harian |
| **Flow Of Events / Urutan Kejadian** | 1. Pengguna E-Diary memberikan kata kunci pada E-Diary. 2. E-Diary mencari catatan harian berdasarkan kata kunci. 3. E-Diary menemukan catatan harian berdasarkan kata kunci. 4. E-Diary menampilkan catatan harian yang dicari. |
| **Alternate Scenario** | - |
| **Exceptional Scenario** | * 1. E-Diary tidak menemukan catatan harian   2. E-Diary memberikan pesan tidak menemukan catatan harian. |

1. Penentuan use case

Use case :

* Menulis catatan harian (dari skenario menulis catatan harian dan menyimpan catatan harian)
* Membaca catatan harian (dari skenario membaca catatan harian)
* Mencari catatan harian (dari mencari catatan harian)

Skenario menulis catatan harian dan menyimpan catatan harian dilebur menjadi satu dengan alasan Pengguna E-Diary menulis catatan harian untuk disimpan.

1. Penentuan hubungan antar use case & use case diagram

Use case menulis catatan harian tidak memiliki hubungan dengan use case yang lain. Use case mencari catatan harian memiliki hubungan dengan membaca catatan harian. Dalam hal ini, ketika Pengguna E-Diary selesai mencari catatan harian, pengguna tersebut memiliki pilihan untuk membaca atau tidak membaca catatan harian yang ditemukan. Berarti, hubungan antara kedua use case tersebut adalah *extend*.



1. Penentuan kebutuhan Non-Fungsional

* Catatan harian hanya dapat disimpan pada E-Diary milik pengguna (*Security*).

Analysis

1. Penentuan objek Entity, Boundary dan Control

* Entity : **Catatan Harian**
* Boundary :
  + **Isian Catatan Harian**, untuk menulis catatan harian
  + **Form Pencarian**, untuk memasukkan kata kunci pencarian

*dapat disatukan fungsinya*

* + **~~Tampilan Hasil Pencarian~~**~~, untuk menampilkan hasil pencarian~~
  + **Tampilan Catatan Harian**, untuk keperluan membaca catatan harian
  + **Daftar Catatan harian**, untuk menampilkan seluruh catatan harian
* Control :
  + **Pencatatan**, untuk mencatat catatan harian
  + **Pencarian,** untuk mencari catatan harian
  + **Pembacaan,** untuk membuka catatan harian

1. Mapping use case ke sequence diagram.

3 use case yang telah dibuat, digunakan untuk menyusun interaksi e, b dan c.

Sequence Diagram mencatat catatan harian

*Objek yang terlibat : Isian Catatan Harian, Pencetakan, Catatan Harian*



Sequence Diagram membaca catatan harian

*Objek yang terlibat : Tampilan Catatan Harian, Daftar Catatan Harian, Pencarian, Pembacaan, Catatan Harian*



Sequence Diagram mencari catatan harian

*Objek yang terlibat : Form Pencarian, Daftar Catatan Harian, Pencarian, Catatan Harian*



1. Memodelkan hubungan antar objek dengan CRC

|  |  |
| --- | --- |
| **Catatan Harian** | |
| Responsibility | Collaboration |
| Memodelkan catatan harian. Bertanggung jawab terhadap data tanggal, isi catatan harian dan tempat diisinya catatan harian. | Pencatatan  Pencarian |

|  |  |
| --- | --- |
| **Isian Catatan Harian** | |
| Responsibility | Collaboration |
| Sebagai antarmuka untuk menulis catatan harian. | Pencatatan |

|  |  |
| --- | --- |
| **Form Pencarian** | |
| Responsibility | Collaboration |
| Sebagai antarmuka untuk memasukkan kata kunci pencarian | Pencarian |

|  |  |
| --- | --- |
| **Tampilan Catatan Harian** | |
| Responsibility | Collaboration |
| Bertanggung jawab untuk menampilkan sebuah catatan harian | Pembacaan |

|  |  |
| --- | --- |
| **Daftar Catatan Harian** | |
| Responsibility | Collaboration |
| Bertanggung jawab untuk menampilkan daftar catatan harian yang telah ada, maupun yang telah ditemukan dari hasil pencarian. | Pencarian  Pembacaan |

|  |  |
| --- | --- |
| **Pencatatan** | |
| Responsibility | Collaboration |
| Bertanggung jawab untuk mencatat catatan yang dituliskan pengguna | Isian Catatan Harian  Catatan Harian |

|  |  |
| --- | --- |
| **Pencarian** | |
| Responsibility | Collaboration |
| Bertanggung jawab untuk mencari catatan yang sudah pernah dituliskan pengguna dan mendapatkan kembali semua catatan harian yang telah dituliskan | Daftar Catatan Harian  Catatan Harian |

|  |  |
| --- | --- |
| **Pembacaan** | |
| Responsibility | Collaboration |
| Bertanggung jawab untuk membaca catatan harian yang telah dituliskan | Daftar Catatan Harian  Tampilan Catatan Harian |

1. Identifikasi hubungan antar objek/kelas

**Pencatatan** dengan **Catatan Harian** memiliki hubungan *dependency* karena Catatan Harian digunakan secara *in brief*. Selebihnya asosiasi. Kelas diagram lengkap adalah dari kesemua kelas yang telah dibuat, ditunjukkan pada gambar di bawah ini :



**Desain**

1. Penentuan tujuan desain

Desain yang dibuat harus mudah untuk menyimpan data pada sistem pengguna.

1. Pendefinisian sub-sistem

Sub sistem berdasarkan use case :

* Mencatat catatan harian
* Mencari catatan harian
* Membaca catatan harian

Pengelompokan kelas pada gambar berikut, berturut-turut dari atas ke bawah, merupakan representasi dari subsistem yang terjadi dari use case yang ada.



Pada gambar di atas, Catatan Harian digunakan oleh ketiga sub sistem. Untuk itu, diperlukan subsistem tersendiri khusus untuk akses data Catatan Harian.



Untuk meminimalkan ketergantungan subsistem, subsistem pencarian dan subsistem pencatatang, maka fungsi untuk mendapatkan seluruh data Catatan Harian didelegasikan ke Subsistem akses data, sehingga subsistem menjadi sebagai berikut :



Sesuai dengan tujuan desain, subsistem akses data harus dapat digunakan dengan mudah untuk mengakses data. Diperlukan tambahan kelas untuk akses data. Sehingga hasil revisi kelas desain adalah sebagai berikut :



1. Mapping sub-sistem ke software/hardware platform

