SISTEM AKADEMIK ASIFA

**Nama Kelompok**

Yehezkiel Windriono (155150200111309)

Jodie Rizky Hidayat (155150200111311)

Faishal Pradipta A. (155150207111187)

SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Nomor Dokumen** | | **Halaman** |
| **SEL01-S06** | | **i/33** |
| **REVISI KE** | **2** | **Tanggal : 29/09/2017** |

REKAYASA PERANGKAT LUNAK | KELAS INFORMATIKA-M

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| **Revisi** | **Deskripsi** |
| **A** | * Memperbaiki use case diagram * Memperbaiki use case scenario |
| **B** | * Memperbaiki use case diagram * Memperbaiki use case scenario * Memperbaiki sequence diagram * Memperbaiki class diagram |
| **C** |  |
| **D** |  |
| **F** |  |
| **G** |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **INDEX**  **TGL** | **A**  10/10/14 | **B**  24/10/14 | **C**  31/10/14 | **D** | **E** | **F** | **G** | **H** |
| Ditulis Oleh |  |  |  |  |  |  |  |  |
| Diperiksa Oleh |  |  |  |  |  |  |  |  |
| Disetujui Oleh |  |  |  |  |  |  |  |  |

DAFTAR HALAMAN PERUBAHAN

|  |  |  |  |
| --- | --- | --- | --- |
| **Halaman** | **Revisi** | **Halaman** | **Revisi** |
|  |  |  |  |

DAFTAR ISI

[DAFTAR PERUBAHAN i](#_Toc463773300)

[DAFTAR HALAMAN PERUBAHAN iii](#_Toc463773301)

[DAFTAR ISI v](#_Toc463773302)

[DAFTAR TABEL vii](#_Toc463773303)

[DAFTAR DIAGRAM ix](#_Toc463773304)

[DAFTAR GAMBAR xi](#_Toc463773305)

[BAB I PENDAHULUAN](#_Toc463773305) 1

[1.1 Deskripsi Umum Dokumen(Ikhtisar)](#_Toc463773305) 1

[1.2 Tujuan Penulisan Dokumen](#_Toc463773305) 1

[1.3 Lingkup Masalah](#_Toc463773305) 1

[1.4 Definisi, Istilah dan Singkatan](#_Toc463773305) 2

[1.5 Aturan Penomoran](#_Toc463773305) 2

[1.6 Referensi](#_Toc463773305) 3

[BAB II DESKRIPSI SISTEM](#_Toc463773305) 4

[2.1 Deskripsi Umum Sistem](#_Toc463773305) 4

[2.2 Deskripsi Umum Perangkat Lunak](#_Toc463773305) 5

[2.3 Fungsi Perangkat Lunak](#_Toc463773305) 6

[2.4 Karakteristik Pengguna](#_Toc463773305) 6

[2.5 Batasan Sistem](#_Toc463773305) 7

[2.6 Lingkungan Operasi](#_Toc463773305) 7

[BAB III DESKRIPSI KEBUTUHAN PERANGKAT LUNAK](#_Toc463773305) 8

[3.1 Identifikasi Kebutuhan Perangkat Lunak](#_Toc463773305) 8

[3.2 Kebutuhan Fungsional dan Non Fungsional](#_Toc463773305) 11

[3.2.1 Kebutuhan Fungsional](#_Toc463773305) 11

[3.2.2 Kebutuhan Non-Fungsional](#_Toc463773305) 12

[3.3 Kebutuhan Antarmuka Eksternal](#_Toc463773305) 13

[3.4 Pemodelan Kebutuhan](#_Toc463773305) 15

[3.2.1 Use Case](#_Toc463773305) 15

[3.2.2 Use Case Scenario](#_Toc463773305) 16

[3.2.3 Sequence Diagram](#_Toc463773305) 24

[3.2.4 Class Diagram](#_Toc463773305) 34

DAFTAR TABEL

Tabel 1.4 Definisi, Istilah, dan Singkatan

Tabel 2.4 Karakteristik Pengguna

[Tabel ‎3.2 Kebutuhan Fungsional dan Non Fungsional](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363)

[Tabel ‎3.2.1 Kebutuhan Fungsional](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362)

[Tabel ‎3.2.1.1 Kebutuhan Fungsional Guest User](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 8

[Tabel ‎3.2.1.2 Kebutuhan Fungsional Member User](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 9

[Tabel ‎3.2.1.3 Kebutuhan Fungsional Admin User](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 10

[Tabel ‎3.2.2 Kebutuhan Non Fungsional](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 12

Tabel 3.4.2 Use Case Scenario

[Tabel 3.4.2.1 Use Case Scenario Login](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 15

[Tabel 3.4.2.2 Use Case Scenario Logout](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 15

[Tabel 3.4.2.3 Use Case Scenario Registrasi Akun](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 16

[Tabel 3.4.2.4 Use Case Scenario Lihat Akun](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 16

[Tabel 3.4.2.5 Use Case Scenario Lihat Member](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 17

Tabel 3.4.2.6 Use Case Scenario Edit Akun... 17

[Tabel 3.4.2.7 Use Case Scenario Nonaktifkan Akun](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 18

[Tabel 3.4.2.8 Use Case Scenario Buat Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 18

[Tabel 3.4.2.9 Use Case Scenario Lihat Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 19

[Tabel 3.4.2.10 Use Case Scenario Edit Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 19

[Tabel 3.4.2.11 Use Case Scenario Hapus Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 20

[Tabel 3.4.2.12 Use Case Scenario Beri Komentar](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 20

[Tabel 3.4.2.13 Use Case Scenario Edit komentar](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 20

[Tabel 3.4.2.14 Use Case Scenario Hapus Komentar](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 21

[Tabel 3.4.2.15 Use Case Scenario Beri Rate](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 21

[Tabel 3.4.2.16 Use Case Scenario Transpose](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 22

Tabel 3.4.2.17 Use Case Scenario Lihat Semua Member 22

Tabel 3.4.2.18 Use Case Scenario Lihat Semua Chord 23

Tabel 3.4.2.19 Use Case Scenario Lihat Semua Komentar 23

Tabel 3.4.2.20 Use Case Scenario Edit Akun Member 24

Tabel 3.4.2.21 Use Case Scenario Edit Chord Member 25

Tabel 3.4.2.22 Use Case Scenario Edit Komentar Member 25

Tabel 3.4.2.23 Use Case Scenario Hapus Akun Member 26

Tabel 3.4.2.24 Use Case Scenario Hapus Chord Member 27

Tabel 3.4.2.25 Use Case Scenario Hapus Komentar Member 27

Tabel 3.4.2.26 Use Case Scenario Pulihkan Akub Member 27

Tabel 3.4.2.27 Use Case Scenario Pulihkan Chord Member 28

Tabel 3.4.2.28 Use Case Scenario Pulihkan Komentar Member 28

DAFTAR DIAGRAM

[Diagram ‎3.4.1 Use Case](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 14

[Diagram ‎3.4.3 Sequence Diagram](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363)

[Diagram ‎3.4.3.1 Sequence Diagram Login](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 29

[Diagram 3.4.3.2 Sequence Diagram Logout](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 30

[Diagram ‎3.4.3.3 Sequence Diagram Registrasi Akun](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448363) 31

[Diagram ‎3.4.3.4 Sequence Diagram Lihat Akun](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 31

[Diagram ‎3.4.3.5 Sequence Diagram Lihat Member](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 32

Diagram 3.4.3.6 Sequence Diagram Edit Akun... 33

[Diagram ‎3.4.3.7 Sequence Diagram Nonaktifkan Akun](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 33

[Diagram ‎3.4.3.8 Sequence Diagram Buat Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 34

[Diagram ‎3.4.3.9 Sequence Diagram Lihat Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 34

[Diagram ‎3.4.3.10 Sequence Diagram Edit Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 35

[Diagram ‎3.4.3.11 Sequence Diagram Hapus Chord](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 36

[Diagram ‎3.4.3.12 Sequence Diagram Beri Komentar](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 36

[Diagram ‎3.4.3.13 Sequence Diagram Edit komentar](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 37

[Diagram ‎3.4.3.14 Sequence Diagram Hapus Komentar](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 37

[Diagram ‎3.4.3.15 Sequence Diagram Beri Rate](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 38

[Diagram ‎3.4.3.16 Sequence Diagram Transpose](file:///C:\Users\Asus\.nbi\downloads\Downloads\Kelompok%203%20-%20SKPL%20-%20PAPEDA%20-%20Rev%200%20(1).docx#_Toc432448362) 38

Diagram ‎3.4.3.17 Sequence Diagram Lihat Semua Member 39

Diagram ‎3.4.3.18 Sequence Diagram Lihat Semua Chord 39

Diagram ‎3.4.3.19 Sequence Diagram Lihat Semua Komentar 40

Diagram ‎3.4.3.20 Sequence Diagram Edit Akun Member 41

Diagram ‎3.4.3.21 Sequence Diagram Edit Chord Member 42

Diagram ‎3.4.3.22 Sequence Diagram Edit Komentar Member 43

Diagram ‎3.4.3.23 Sequence Diagram Hapus Akun Member 43

Diagram ‎3.4.3.24 Sequence Diagram Hapus Chord Member 44

Diagram ‎3.4.3.25 Sequence Diagram Hapus Komentar Member 44

Diagram ‎3.4.3.26 Sequence Diagram Pulihkan Akub Member 45

Diagram ‎3.4.3.27 Sequence Diagram Pulihkan Chord Member 45

Diagram ‎3.4.3.28 Sequence Diagram Pulihkan Komentar Member 46

Diagram 3.4.4 Class Diagram 47

DAFTAR GAMBAR

Gambar 2.1 Arsitektur Aplikasi CHORON 4

# PENDAHULUAN

Dokumen ini berisikan Spesifikasi Kebutuhan Perangkat Lunak (SKPL) atau Software Requirement Spesification (SRS) Untuk CHORON (Chord Online).

## Deskripsi Umum Dokumen

Dokumen SKPL ini dibagi menjadi tiga bagian utama. Bagian pertama berisi Pendahuluan, bagian kedua berisi Deskripsi Sistem, dan bagian ketiga berisi Deskripsi Kebutuhan Perangkat Lunak.

Pada bagian pertama yaitu Pendahuluan berisikan Deskripsi Umum Dokumen, Tujuan Penulisan dari Dokumen, Lingkup Masalah yang diselesaikan oleh perangkat lunak, definisi dan referensi yang digunakan dalam pengembangan perangkat lunak.

Pada Bagian kedua yaitu Deskripsi Sistem berisikan deskripsi umum sistem, deskripsi umum perangkat lunak, karakteristik pengguna,batasan sistem,lingkungan operasi

Pada Bagian ketiga yaitu Deskripsi Kebutuhan Perangkat Lunak yang berisikan identifikasi kebutuhan perangkat lunak, kebutuhan fungsional dan non fungsional, pemodelan kebutuhah beserta diagram yang sesuai dengan pemodelan kebutuhan. Pada bagian ketiga ini kebutuhan perangkat lunak diuraikan secara lebih rinci.

## Tujuan Penulisan Dokumen

Tujuan dari penulisan dokumen ini adalah untuk menjelaskan deskripsi secara lebih detail yang berkaitan dengan pengembangan perangkat lunak CHORON (Chord Online) serta untuk menspesifikasikan kebutuhan system yang akan dibuat dan digunakan sebagai acuan dalam tahapan pengembangan system

## Lingkup Masalah

CHORON (Chord Online) merupakan sebuah perangkat lunak berbasis website application yang digunakan untuk membantu user dalam hal mencari informasi tentang Chord secara online, setelah itu software yang berbasis website ini dapat diakses dengan menggunakan PC/Komputer dan Laptop yang tersambung ke jaringan internet.

User dapat melakukan registrasi untuk bisa login ke sistem yang dikelola oleh User dalah hal daftar user yang telah melakukan registrasi, kemudian setelah itu user dapat menggunakan software tersebut

Selain itu, Software ini harus terhubung ke jaringan internet agar dapat menampilkan seluruh isi informasi yang ada yang diinginkan oleh user.software ini secara umum dapat menampilkan Chord-chord yang telah dipost oleh user, Tidak hanya mengetahui sebuah informasi mengenai Chord-chord, user juga dapat memberikan komentar dan rating pada Chord yang telah dibuat oleh user lain.

## Definisi, Istilah, dan Singkatan

|  |  |
| --- | --- |
| Istilah dan Akronim | Uraian |
| SKPL | Spesifikasi Kebutuhan Perangkat Lunak, atau dalam bahasa Inggris-nya sering juga disebut sebagai Software Requirements Spesification (SRS), dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan. |
| CHORON | Kepanjangan dari Chord Online |
| Chord | Chord adalah kumpulan tiga nada atau lebih yang bila dimainkan secara bersamaan. |
| DBMS | Database Management Sistem, yaitu suatu sistem yang bertugas untuk menyimpan dan memanipulasi data yang ada. |
| OS | Operating sistem, yaitu suatu perangkat lunak yang menghubungkan interaksi antara pengguna dengan perangkat keras. |
| PL | Merupakan singkatan dari perangkat lunak |
| User | Dibedakan menjadi 2 yaitu Member dan Guest |
| Hardware | Perangkat keras seperti Laptop,PC,Hp dll |
| Transpose | Menaik-turunkan kunci dasar pada suatu lagu |

Tabel 1.4 Definisi, Istilah, dan Singkatan

## Aturan Penomoran

Kode : SiakASF-F-001

Nomor urut representasi kebutuhan mulai dari 01

Jenis representasi kebutuhan,

F : Kebutuhan Fungsional

NF : Kebutuhan Non Fungsional

Singkatan dari nama perangkat lunak

## Referensi

Modul RPL

Dokumen SKPL yang sudah ada sebelumnya sebagai referensi

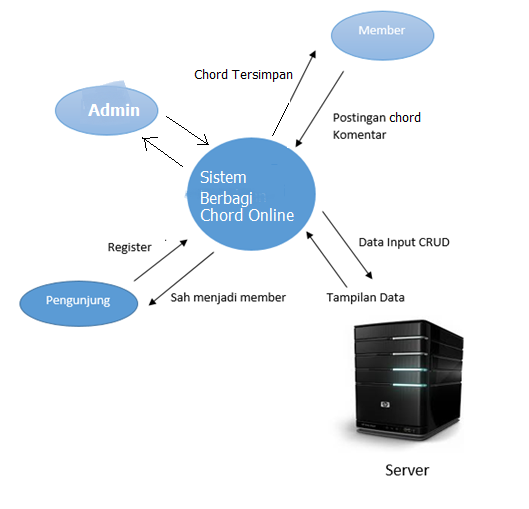
*http://www.tutorialspoint.com/software\_engineering/*

*http://www.uml-diagrams.org/*

# DESKRIPSI SISTEM

## Deskripsi Umum Sistem

CHORON atau Chord Online merupakan sebuah sistem aplikasi yang berbasis web application yang bertujuan untuk mengetahui segala macam Chord yang dibuat oleh user website itu sendiri. Sebuah informasi yang dibutuhkan terkait dengan kendala yang dihadapi, dengan adanya sistem ini mungkin seseorang dapat lebih terbantu dan juga sangat efektif ketika ingin bermain lagu tetapi tidak tahu chord untuk memainkannya, cara kerja atau gambaran dari sistem adalah sebagai berikut :



Gambar 2.1 Arsitektur CHORON

## Deskripsi Umum Perangkat Lunak

CHORON adalah sebuah perangkat lunak berbasis website application yang digunakan untuk membantu user dalam hal mencari informasi tentang Chord lagu secara online. Sebelum menggunakan CHORON, pengunjung harus melakukan proses registrasi terlebih dahulu. User dapat melakukan registrasi untuk bisa login ke sistem yang dikelola oleh User adalah hal daftar user yang telah melakukan registrasi.

Selain sebagai sarana tukar pendapat Chord, CHORON juga menyediakan berita dan Chord terkini mengenai Perkembangan music terkini.

Hal yang bisa dilakukan oleh semua aktor yang terlibat dalam sebuah website ini adalah sebagai berikut :

* User (Member dan Guest) dapat melihat Chord, tetapi hanya pengunjung yang tidak bisa membuat sebuah Chord di website tersebut. Penyebabnya adalah pengunjung belum terdaftar sebagai member untuk membuat Chord.
* Hanya member yang dapat memberikan komentar dalam setiap chord yang telah dipost di website tersebut. Komentar itu dapat dihapus apabila sudah terlanjur mengisi komentar yang tidak semestinya. Komentar juga dapat diedit apabila salah mengisi komentar tersebut. Komentar yang sudah diisi oleh user dapat dilihat oleh user. Apabila user tidak melengkapi isi komentar ketika tekan Submit, maka akan muncul sebuah peringatan bahwa komentar harus diisi
* Untuk mempermudah pencarian sebuah Chord di website, telah disediakan search engine yang berfungsi mencari kata kunci atau genre yang berkaitan dengan Chord tersebut. Setelah pencarian di search engine, maka akan muncul banyak Chord yang berkaitan dengan kata kunci tersebut.

## Fungsi Produk/Perangkat Lunak

Pada subbab ini akan menjelaskan fungsi-fungsi yang dapat diakses oleh pengguna website CHORON, tetapi tidak dijelaskan secara spesifik. Untuk detail selengkapnya akan dijelaskan pada bab III. Adapun fungsi-fungsi yang dimiliki oleh perangkat lunak ini adalah:

* Member User
* Membuat Chord
* Membuat komentar
* Mendapat informasi dari Chord yang ada
* Mencari Chord melalui search engine
* Guest User
* Mendapat informasi dari Chord yang ada
* Mencari Chord melalui search engine

## Karakteristik Pengguna

Tabel 2.4 Karakteristik Pengguna

|  |  |  |
| --- | --- | --- |
| **No** | **Identifikasi Pengguna** | **Karakteristik** |
| 1 | Member | * Melakukan create, read, update, delete terhadap komentar dan Chord * Melakukan pencarian kata kunci yang berkaitan dengan Chord |
|  |  |  |
|  |  |  |

## Batasan

CHORON memiliki batasan – batasan dalam penggunaannya sebagai berikut:

* Pengguna harus terhubung ke internet
* Terikat dengan software/browser
* Perangkat lunak ini dapat dikembangkan dengan bahasa pemrograman web (PHP, HTML, dan Javascript)
* DBMS yang digunakan adalah MySQL
* Perangkat lunak harus terhubung dahulu ke jaringan internet agar bisa digunakan
* Browser yang digunakan dalam website adalah Mozilla Firefox, Google Chrome, Safari, Internet Explorer, dan Opera
* OS yang digunakan adalah Windows (XP, Vista, 7, 8, 10) dan Linux (Ubuntu, Mint)
* Semua berhak membuat Chord dengan syarat sudah terdaftar sebagai member user dalam website ini baik untuk pribadi maupun komunitas.
* Chord dapat ditampilkan di website apabila sudah mendapat ACC dari seorang User

## Lingkungan Operasi

Lingkungan operasi dari SIF sebagai berikut :

Server

* Web application
* Web server
* Database
* OS (Windows Server)
* Hardware (PC/Laptop) input device (keyboard,mouse)

Client

* Browser
* OS (Windows)
* Hardware (PC/Laptop) input device (keyboard,mouse)

# DESKRIPSI KEBUTUHAN PERANGKAT LUNAK

## Identifikasi Kebutuhan PL

Seiring perkembangan teknologi, masyarakat tentunya perlu membutuhkan sesuatu yang bisa mempermudah mereka dalam mendapatkan sebuah informasi, berkaca dari permasalahan tentang pembuatan CHORON ini bahwa masyarakat masih kesulitan dalam mendapatkan sebuah informasi utamanya pada saat keadaan mendesak, dengan adanya CHORON ini mungkin bisa membantu memudahkan masyarakat dalam mendapatkan informasi khususnya di bidang penulisan chord.

Berdasarkan observasi yang ada disekitar lingkungan kami, sebagian masyarakat kesulitan dalam menentukan chord saat mengiringi atau memainkan lagu, dan ada juga masyarakat yang mempunyai bakat untuk menulis chord namun tidak memiliki media untuk menuliskannya. Hal ini menjadi kesempatan jika ada media yang menghubungkan dua golongan ini. Sehingga jika ada masyarakat yang membutuhkan chord dapat memainkan lagu yang diinginkan dan masyarakat yang pandai menulis chord dapat menunjukkan kemampuannya.

Untuk menyelesaikan permasalahan tersebut, website ini menyediakan fitur-fitur yang akan dibahas dalam dokumen ini. Antara lain CRUD Chord, CRUD komentar, dan search engine. CRUD Chord memiliki peran yaitu membuat Chord, menghapus Chord, mengedit Chord, dan menampilkan Chord. CRUD komentar memiliki peran yaitu membuat komentar, menghapus komentar, mengedit komentar, dan melihat komentar. Komentar yang dibuat oleh user lain dapat menggambarkan sebuah penilaian dari user lain berupa teks dan rating. Search engine berguna untuk mencari banyak Chord melalui sebuah kata kunci.

## Kebutuhan Fungsional dan Non Fungsional

### Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode** | **Nama Fungsi** | **Deskripsi** |
| 1 | SAAsf-F-001 | Login | Member user dapat melakukan login |
| 2 | SAAsf -F-002 | Lihat Keuangan | Member user dapat melakukan login |
| 3 | SAAsf -F-003 | Lihat Rapor | Member user dapat keluar dari akun yang sedang login |
| 4 | SAAsf -F-004 | Lihat Kehadiran | Member user dapat melihat profil akunnya sendiri |
| 5 | SAAsf -F-005 | Lihat Tugas | Member user dapat melihat profil member lain |
| 6 | SAAsf -F-006 | Lihat Profil | Member user mengedit profil akunnya |
| 7 | SAAsf -F-007 | Pengaturan | Member user dapat menonaktifkan akunnya |
| 8 | SAAsf -F-008 | Buat Chord | Member user dapat memposting Chord di website |
| 9 | SAAsf -F-009 | Lihat Chord | Member user dapat melihat Chord yang telah diposting |

### Kebutuhan Non Fungsional

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Kode** | **Nama Fungsi** | **Parameter** | **Deskripsi** |
| 1. | CHORON-NF-02 | Akses pengguna | Accessibility | Aplikasi ini dapat digunakan di berbagai browser |
| 2. | CHORON-NF-03 | Keamanan User | Security | Aplikasi ini dilengkapi dengan kemanan password User dengan menggunakan Hash, agar data terlindungi dan tidak dapat dimanipulasi oleh pihak yang tidak berwenang. |

Tabel 3.2.2 Daftar Kebutuhan Non Fungsional

## Kebutuhan Antarmuka Eksternal

### Antarmuka Pengguna

Pengguna mampu melihat aplikasi yang digunakan pada layar monitor. Untuk mengoperasikan aplikasi yang digunaka, pengguna menggunakan keyboard dan mouse.

### Antarmuka Perangkat Keras

Kebutuhan minimum perangkat keras yang dibutuhkan pada aplikasi SIF (Sistem Informasi Farmasi) adalah :

1. PC
2. Mouse
3. Layar Monitor
4. Keyboard
5. Graphic card
6. Ethernet card
7. Modem

### Antarmuka Perangkat Lunak

Kebutuhan perangkat lunak yang dibutuhkan pada aplikasi CHORON (Sharing Informasi Farmasi) menggunakan perangkat lunak Web Browser dan untuk tampilannya menggunakan CSS.

Website memerlukan beberapa aplikasi pendukung untuk menjalankan sebuah website tersebut, antara lain:

* Web Server Apache dan database MySQL
* Browser Google Chrome
* OS Windows 7

### Antarmuka Komunikasi

Yang dibutuhkan hanya sebuah komputer *server* dan beberapa komputer *client (sesuai jumlah kebutuhan)* yang terhubung secara *client-server* dalam lingkup jaringan Internet atau intranet berbasisprotokol HTTP. Antarmuka komunikasi yang dibutuhkan dalam perangkat lunak ini merupakan antarmuka untuk melakukan koneksi dalam jaringan internet meliputi:

1. Antarmuka komunikasi pada sisi Server

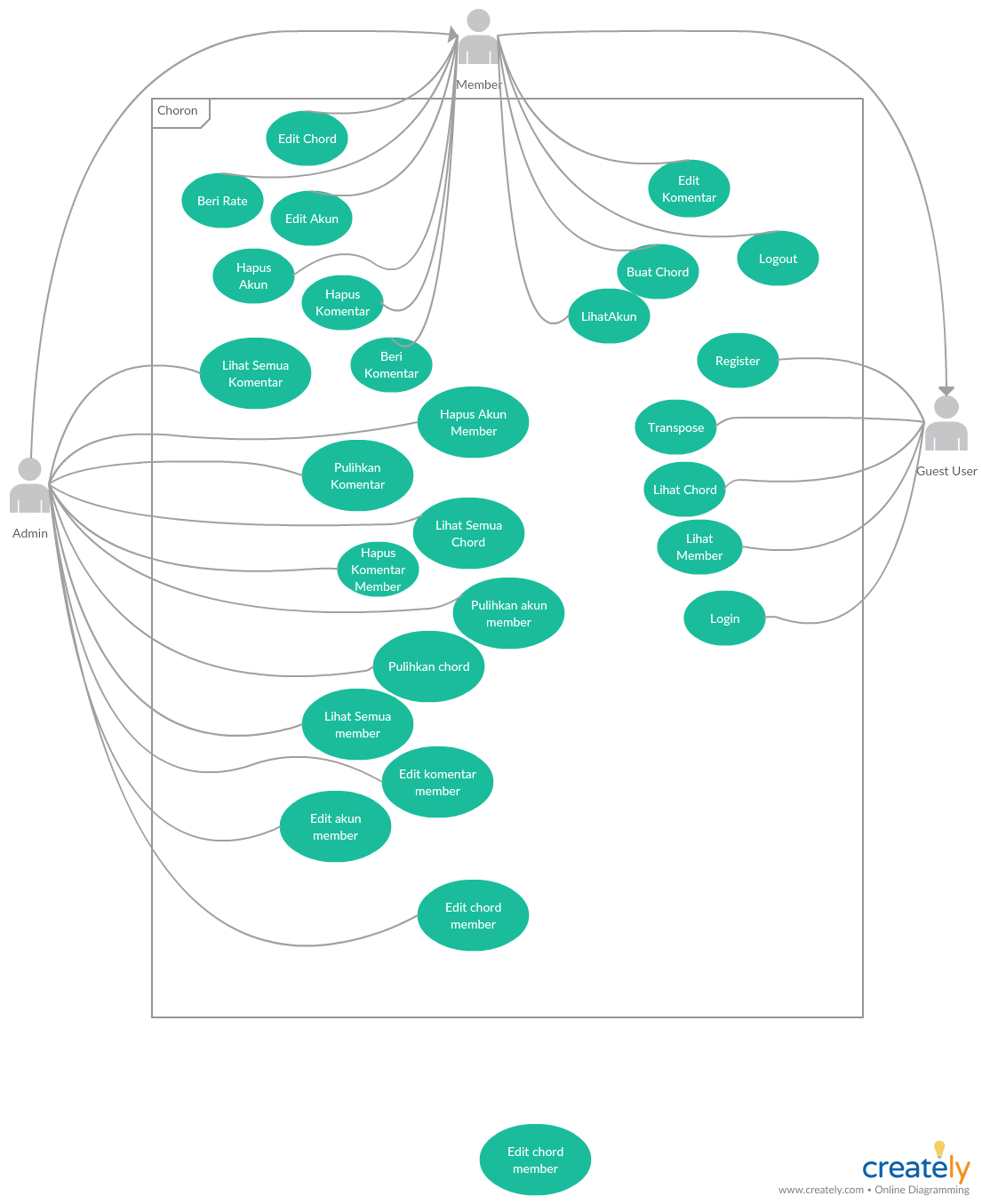
Pada sisi server dibutuhkan adanya sebuah web server yang terhubung dalam jaringan internet. Web server tersebut harus mempunyai alamat IP dan domain yang dapat dikenali dalam jaringan internet.

1. Antarmuka komunikasi pada sisi Client

Pada sisi client proses yang dilakukan adalah melakukan request kepada server untuk meminta layanan http. Oleh karena itu antarmuka yang dibutuhkan pada sisi client adalah perangkat komunikasi yang memungkinkan komputer client terhubung dengan jaringan internet seperti wifi atau jaringan internet lain dengan kecepatan minimal 256 kbps.

## Pemodelan Kebutuhan

### Use Case



Tabel 3.4.1 Use Case Diagram

Berdasarkan gambar di atas, terdapat 3 aktor yang melibatkan aktivitas di sistem website tersebut, yaitu User, member user, dan Guest user atau pengunjung. Member user dapat melakukan CRUD Chord, CRUD Komentar, CRUD akun (kecuali register akun), Login, Logout, dan pencarian kata kunci melalui search engine. Guest User atau Pengunjung dapat melakukan CRUD Akun, CRUD Komentar, Lihat Chord, dan pencarian kata kunci. User dapat melakukan semua aktivitas di website ini, termasuk seleksi Chord.

### Use Case Scenario

**Tabel 3.4.2.1 Use Case Scenario Login**

|  |  |
| --- | --- |
| CHORON-F-001 | Login |
| Objective | Aktor ingin melakukan login |
| Actors | Member dan Admin |
| Pre-condition | Aktor sudah masuk ke halaman login |
| Main flow | Aktor memasukkan username dan password pada form  Sistem melakukan pengecekan database pada inputan user |
| Alternative flow | Jika aktor tidak mengisi form username dan password maka sistem akan menampilkan pesan bahwa user harus mengisi form username dan password.  Jika username user salah/tidak terdaftar maka sistem akan menampilkan pesan bahwa username atau password salah. |
| Post-condition | Aktor telah masuk/login ke dalam sistem dengan akun milik user sendiri |

Tabel 3.4.2.1 Use Case Scenario Login

**Tabel 3.4.2.2 Use Case Scenario Logout**

|  |  |
| --- | --- |
| CHORON-F-02 | Logout |
| Objective | Aktor ingin melakukan logout |
| Actors | Member dan Admin |
| Pre-condition | Aktor sudah melakukan login ke system |
| Main flow | 1. Aktor melakukan klik pada tombol logout 2. Sistem menghapus session dari aktor yang bersangkutan 3. Sistem akan mengirim aktor ke halaman utama |
| Alternative flow | - |
| Post-condition | Session dari aktor akan berakhir/dihapus dan aktor berhasil melakukan logout |

**Tabel 3.4.2.3 Use Case Scenario Registerasi Akun**

|  |  |
| --- | --- |
| CHORON-F-003 | Registerasi Akun |
| Objective | Aktor ingin melakukan pendaftaran/registrasi |
| Actors | Guest User |
| Pre-condition | Aktor telah melakukan klik pada tombol Daftar |
| Main flow | 1. Aktor melakukan pengisian pada formulir yang terdiri atas username, email dan password. 2. Aktor melakukan klik pada tombol daftar.   Sistem mengirimkan data formulir ke database.   1. Sistem mengirimkan pesan kepada aktor bahwa pendaftaran berhasil dan aktor dapat login menggunakan username dan passwordnya. |
| Alternative flow | 1. Jika inputan user terdapat htmlspecialchars maka user akan dikembalikan ke halaman registrasi.  2. Jika user tidak memasukkan username, email atau password sistem akan memunculkan pesan kolom harus diisi.  3. Jika username user sudah terdapat di database, program akan mengembalikan user ke halaman register dengan pesan “username telah dipakai” |
| Post-condition | Aktor telah mendapatkan informasi akun yang akan digunakan untuk melakukan login. |

Tabel 3.4.2.2 Use Case Scenario Registrasi Akun

**Tabel 3.4.2.4 Use Case Scenario Lihat Akun**

|  |  |
| --- | --- |
| CHORON-F-004 | Lihat Akun |
| Objective | Aktor ingin melihat informasi akun miliknya |
| Actors | Member dan Admin |
| Pre-condition | Aktor telah melakukan pendaftaran dan login |
| Main flow | 1. Aktor melakukan klik pada tombol profilnya 2. Sistem membawa aktor ke halaman akun |
| Alternative flow | - |
| Post-condition | Aktor telah berhasil melihat informasi akun miliknya |

**Tabel 3.4.2.5 Use Case Scenario Lihat Member**

|  |  |
| --- | --- |
| CHORON-F-005 | Lihat Member |
| Objective | Aktor ingin melihat informasi akun miliknya |
| Actors | Guest, Member dan Admin |
| Pre-condition | Aktor melihat username member yang telah terdaftar |
| Main flow | 1. Aktor melakukan klik pada tombol username member 2. Sistem membawa aktor ke halaman member yang bersangkutan |
| Alternative flow | - |
| Post-condition | Aktor telah berhasil melihat informasi member yang terdaftar |

**Tabel 3.4.2.6 Use Case Scenario Edit Akun**

|  |  |
| --- | --- |
| CHORON-F-006 | Edit Akun |
| Objective | Aktor ingin melakukan perubahan pada akun |
| Actors | Member dan Admin |
| Pre-condition | Aktor telah melakukan login |
| Main flow | 1. Aktor melakukan klik pada tombol edit akun 2. Sistem membawa aktor ke halaman form edit akun 3. Aktor melakukan perubahan pada halaman for edit akun 4. Aktor melakukan klik pada tombol simpan 5. Sistem melakukan validasi dan menyimpan perubahan yang dilakukan oleh aktor |
| Alternative flow | 1. Jika aktor membatalkan pengeditan akun melalui tombol batal, maka akan kembali ke halaman utama. |
| Post-condition | Aktor telah berhasil melakukan perubahan pada akun |

Tabel 3.4.2.6 Use Case Scenario Edit Akun

**Tabel 3.4.2.7 Use Case Scenario Nonaktifkan Akun**

|  |  |
| --- | --- |
| CHORON-F-007 | Nonaktifkan Akun |
| Objective | Aktor ingin menghapus akun |
| Actors | Member dan Admin |
| Pre-condition | Aktor telah melakukan login |
| Main flow | 1. Aktor melakukan klik pada tombol edit akun 2. Sistem membawa aktor ke halaman edit akun 3. Aktor melakukan klik pada tombol nonaktifkan akun 4. Sistem menampilkan kotak dialog konfirmasi 5. Aktor melakukan klik pada tombol iya 6. Sistem melakukan perubahan status akun pada database |
| Alternative flow | Jika aktor ingin membatalkan aksi, maka sistem akan mengembalikan aktor ke halaman akun |
| Post-condition | Akun user telah resmi dinonaktifkan |

Tabel 3.4.2.7 Use Case Scenario Nonaktifkan Akun

**Tabel 3.4.2.8 Use Case Scenario Membuat Chord**

|  |  |
| --- | --- |
| CHORON-F-008 | Membuat Chord |
| Objective | Aktor ingin membuat Chord |
| Actors | member user |
| Pre-condition | Aktor harus sudah melakukan login |
| Main flow | 1. Aktor melakukan klik pada tombol buat Chord 2. Sistem mengirim aktor ke halaman editor 3. Aktor membuat Chord dan melakukan simpan 4. Sistem mengirimkan Chord ke halaman seleksi Chord |
| Alternative flow | 1. Jika aktor tidak mengisi data Chord dengan lengkap, maka Chord dapat masuk kedalam draft melalui tombol simpan. 2. Jika aktor membatalkan pengisian Chord melalui tombol batal, maka akan kembali ke halaman utama website. |
| Post-condition | Aktor menunggu kepastian Chord yang akan ditampilkan di website. |

Tabel 3.4.2.8 Use Case Scenario Membuat Chord

**Tabel 3.4.2.9 Use Case Scenario Lihat Chord**

|  |  |
| --- | --- |
| CHORON-F-009 | Lihat Chord |
| Objective | Aktor ingin melihat Chord |
| Actors | Guest, Member, dan Admin |
| Pre-condition | Aktor masuk ke halaman utama |
| Main flow | 1. Aktor melakukan klik pada judul Chord. 2. Sistem menampilkan Chord. |
| Alternative flow | - |
| Post-condition | Aktor telah berhasil melihat Chord |

Tabel 3.4.2.8 Use Case Scenario Lihat Chord

**Tabel 3.4.2.10 Use Case Scenario Edit Chord**

|  |  |
| --- | --- |
| CHORON-F-010 | Edit Chord |
| Objective | Aktor ingin melakukan perubahan pada Chord |
| Actors | Member user dan Admin |
| Pre-condition | Aktor sudah melakukan login. Aktor sudah membuat Chord |
| Main flow | 1. Aktor melakukan pengeditan pada form Chord 2. Aktor menekan tombol simpan 3. Sistem menyimpan perubahan 4. Aktor telah selesai melakukan edit |
| Alternative flow | 1. Jika aktor membatalkan pengeditan Chord melalui tombol batal, maka akan kembali ke halaman chord yang bersangkutan. |
| Post-condition | Aktor berhasil melakukan perubahan pada Chord |

Tabel 3.4.2.9 Use Case Scenario Edit Chord

**Tabel 3.4.2.11 Use Case Scenario Hapus Chord**

|  |  |
| --- | --- |
| CHORON-F-011 | Hapus Chord |
| Objective | Aktor ingin melakukan penghapusan pada Chord yang sudah diposting |
| Actors | member user dan Admin |
| Pre-condition | Aktor telah memiliki Chord |
| Main flow | 1. Aktor melakukan klik pada tombol edit Chord 2. Aktor melakukan klik pada tombol hapus chord 3. Sistem memunculkan dialog box 4. Aktor mengklik ya. 5. Sistem melakukan perubahan status Chord pada database 6. Sistem meredirect actor ke halaman utama |
| Alternative flow | 1. Jika aktor memilih untuk membatalkan, maka sistem mengembalikan aktor ke halaman akun |
| Post-condition | Chord telah berhasil dihapus |

Tabel 3.4.2.10 Use Case Scenario Hapus Chord

**Tabel 3.4.2.12 Use Case Scenario Beri Komentar**

|  |  |
| --- | --- |
| CHORON-F-012 | Beri Komentar |
| Objective | Aktor ingin berkomentar terhadap sebuah Chord |
| Actors | Member, dan Admin |
| Pre-condition | Aktor sudah melakukan login, dan lihat Chord |
| Main flow | 1. Aktor mengetikkan komentar 2. Aktor melakukan klik komentar 3. Sistem menyimpan komentar tersebut 4. Sistem meridirect ke halaman chord yang bersangkutan |
| Alternative flow | 1. Jika komentar tidak diisi, maka muncul sebuah peringatan bahwa data harus diisi dalam komentar tersebut. |
| Post-condition | Komentar dari aktor telah ditampilkan |

Tabel 3.4.2.11 Use Case Scenario Tambah Komentar

**Tabel 3.4.2.13 Use Case Scenario Edit Komentar**

|  |  |
| --- | --- |
| CHORON-F-012 | Edit Komentar |
| Objective | Aktor ingin melakukan edit pada komentar |
| Actors | Member, dan Admin |
| Pre-condition | Aktor telah melakukan login, buat komentar, dan lihat chord |
| Main flow | 1. Aktor melakukan klik pada tombol edit komentar 2. Sistem menampilkan halaman form edit komentar 3. Aktor mengedit komentar dan melakukan klik pada tombol simpan perubahan 4. Sistem menyimpan perubahan |
| Alternative flow | 1. Jika komentar tidak diisi dengan lengkap, maka muncul sebuah peringatan bahwa data harus diisi dalam komentar tersebut. |
| Post-condition | Aktor telah melakukan edit pada komentar |

Tabel 3.4.2.13 Use Case Scenario Edit Komentar

**Tabel 3.4.2.14 Use Case Scenario Hapus Komentar**

|  |  |
| --- | --- |
| CHORON-F-013 | Hapus Komentar |
| Objective | Aktor ingin menghapus komentar |
| Actors | Member , dan Admin |
| Pre-condition | Aktor sudah melakukan login  Aktor sudah melakukan lihat Chord  Aktor sudah melakukan Buat komentar |
| Main flow | 1. Aktor melakukan klik pada tombol hapus komentar 2. Sistem menampilkan dialog box 3. Aktor melakukan klik pada tombol ya 4. Sistem mengubah status keaktifan komentar tersebut |
| Alternative flow | 1. Jika aktor memilih untuk membatalkan hapus komentar, maka sistem tidak menghapus komentar tersebut. |
| Post-condition | Komentar dari aktor telah dihapus |

Tabel 3.4.2.13 Use Case Scenario Hapus Komentar

**Tabel 3.4.2.15 Use Case Scenario Beri Rate**

|  |  |
| --- | --- |
| CHORON-F-015 | Beri Rate |
| Objective | Aktor ingin memberi rating pada chord |
| Actors | Member, dan Admin |
| Pre-condition | Aktor sudah melakukan login, dan lihat Chord |
| Main flow | 1. Aktor memilih tingkatan tingkatan rating 2. Aktor menekan tombol rate   Sistem mengirim data rating ke database |
| Alternative flow | - |
| Post-condition | Sistem menampilkan chord yang sudah ditranspose |

Tabel 3.4.2.16 Use Case Scenario Logout

**Tabel 3.4.2.16 Use Case Scenario Transpose**

|  |  |
| --- | --- |
| CHORON-F-016 | Transpose |
| Objective | Aktor ingin melakukan Transpose |
| Actors | Member, Guest, dan Admin |
| Pre-condition | Aktor sudah melakukan lihat Chord |
| Main flow | 1. Aktor memilih tingkatan transpose 2. Aktor menekan tombol transpose 3. Sistem meredirect ke halaman chord dan mengubah keynote pada postingan Chord yang bersangkutan |
| Alternative flow | - |
| Post-condition | Sistem menampilkan chord yang sudah ditranspose |

Tabel 3.4.2.16 Use Case Scenario Logout

**Tabel 3.4.2.17 Use Case Scenario Melihat Semua akun member**

|  |  |
| --- | --- |
| CHORON-F-017 | Melihat Semua Akun Member |
| Objective | Aktor ingin melihat semua member yang terdaftar |
| Actors | Admin |
| Pre-condition | Aktor sudah melakukan login dan berada di homepage admin |
| Main flow | 1 .Aktor mengklik lihat semua member  2. Sistem mengambil data member dari database  3. Sistem meredirect actor ke halaman lihat semua member |
| Alternative flow | - |
| Post-condition | Sistem menampilkan semua member yang ada di database |

Tabel 3.4.2.16 Use Case Scenario Lihat Semua Akun Member

**Tabel 3.4.2.18 Use Case Scenario Melihat Semua Chord Member**

|  |  |
| --- | --- |
| CHORON-F-018 | Melihat Semua Chord Member |
| Objective | Aktor ingin melihat semua chord yang telah diposting member |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada dashboard admin |
| Main flow | 1. Aktor melakukan klik Lihat Semua Chord 2. Sistem mengambil data dari database 3. Sistem me-redirect actor ke halaman lihat semua chord |
| Alternative flow | - |
| Post-condition | Sistem menampilkan semua chord yang diposting pada website |

Tabel 3.4.2.18 Use Case Scenario Lihat Semua Chord Member

**Tabel 3.4.2.19 Use Case Melihat Semua Komentar Member**

|  |  |
| --- | --- |
| CHORON-F-019 | Melihat Semua Komentar Member |
| Objective | Aktor ingin melihat semua komentar ada pada website. |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada dashboard Admin |
| Main flow | 1. Aktor melakukan klik Lihat Semua Komentar. 2. Sistem mengambil data dari database. 3. Sistem me-redirect actor ke halaman lihat semua komentar. |
| Alternative flow |  |
| Post-condition | Sistem menampilkan semua komentar yang ada pada website. |

Tabel 3.4.2.19 Use Case Scenario Lihat Semua Komentar Member

**Tabel 3.4.2.20 Use Case Edit Akun Member**

|  |  |
| --- | --- |
| CHORON-F-020 | Edit Akun Member |
| Objective | Aktor ingin melakukan perubahan informasi member yang telah terdaftar |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Member |
| Main flow | 1. Aktor melakukan klik Edit pada salah satu member 2. Sistem mengambil informasi member tersebut dari database. 3. Sistem me-redirect actor ke halaman AdminFormEditMember. 4. Aktor melakukan perubahan informasi dengan menginput informasi yang telah diubah /diinginkan pada form-form yang tersedia. 5. Aktor melakukan klik Konfirmasi 6. Sistem menginputkan semua informasi tersebut ke dalam database 7. Sistem me-redirect actor ke halaman Lihat Semua Member |
| Alternative flow | Jika setelah melakukan klik tombol konfirmasi dan ternyata ada informasi yang tidak valid( semisal penulisan email yang tidak sesuai), system tidak akan menginputkan semua informasi tersebut ke dalam database dan me-redirect ke halaman error 404 |
| Post-condition | Sistem menampilkan akun member dengan informasi yg telah diubah. |

Tabel 3.4.2.20 Use Case Scenario Edit Akun Member

**Tabel 3.4.2.21 Use Case Edit Chord Member**

|  |  |
| --- | --- |
| CHORON-F-021 | Edit Chord Member |
| Objective | Aktor ingin melakukan perubahan chord member yang telah dipost |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Chord |
| Main flow | 1. Aktor melakukan klik Edit pada salah satu chord 2. Sistem mengambil informasi member tersebut dari database. 3. Sistem me-redirect actor ke halaman AdminFormEditChord. 4. Aktor melakukan perubahan informasi dengan menginput informasi yang telah diubah /diinginkan pada form-form yang tersedia. 5. Aktor melakukan klik tombol Submit 6. Sistem menginputkan semua informasi tersebut ke dalam database 7. Sistem me-redirect actor ke halaman Lihat Semua Chord |
| Alternative flow |  |
| Post-condition | Sistem menampilkan chord dengan informasi yg telah diubah. |

Tabel 3.4.2.21 Use Case Scenario Edit Chord Member

**Tabel 3.4.2.22 Use Case Edit Komentar Member**

|  |  |
| --- | --- |
| CHORON-F-022 | Edit Komentar Member |
| Objective | Aktor ingin melakukan perubahan komentar yang telah dipost |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Komentar |
| Main flow | 1. Aktor melakukan klik Edit pada salah satu member 2. Sistem mengambil informasi member tersebut dari database. 3. Sistem me-redirect actor ke halaman AdminFormEditKomentar. 4. Aktor melakukan perubahan informasi dengan menginput informasi yang telah diubah /diinginkan pada form-form yang tersedia. 5. Aktor melakukan klik Konfirmasi 6. Sistem menginputkan semua informasi tersebut ke dalam database 7. Sistem me-redirect actor ke halaman Lihat Semua Komentar |
| Alternative flow | - |
| Post-condition | Sistem menampilkan komentar yg telah diubah. |

Tabel 3.4.2.22 Use Case Scenario Edit Komentar Member

**Tabel 3.4.2.23 Use Case Hapus Akun Member**

|  |  |
| --- | --- |
| CHORON-F-023 | Hapus Akun Member |
| Objective | Aktor ingin melakukan penghapusan terhadap akun member |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Member |
| Main flow | 1. Aktor melakukan klik Hapus pada salah satu member 2. Sistem memindahkan status member tersebut menjadi nonaktif |
| Alternative flow | - |
| Post-condition | Member terdapat di dalam table member yang telah terhapus dan member telah hilang dari website |

Tabel 3.4.2.23 Use Case Scenario Hapus Akun Member

**Tabel 3.4.2.24 Use Case Hapus Chord Member**

|  |  |
| --- | --- |
| CHORON-F-024 | Hapus Chord Member |
| Objective | Aktor ingin melakukan penghapusan terhadap chord |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Chord |
| Main flow | 1. Aktor melakukan klik Hapus pada salah satu chord 2. Sistem memindahkan status chord tersebut menjadi nonaktif |
| Alternative flow | - |
| Post-condition | Chord terdapat di dalam table chord yang telah terhapus dan chord telah hilang dari website |

Tabel 3.4.2.24 Use Case Scenario Hapus Chord Member

**Tabel 3.4.2.25 Use Case Hapus Komentar Member**

|  |  |
| --- | --- |
| CHORON-F-025 | Hapus Komentar Member |
| Objective | Aktor ingin melakukan penghapusan terhadap komentar |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Komentar |
| Main flow | 1. Aktor melakukan klik Hapus pada salah satu komentar 2. Sistem memindahkan status komentar tersebut menjadi nonaktif |
| Alternative flow | - |
| Post-condition | Komentar terdapat di dalam table komentar yang telah terhapus dan komentar telah hilang dari website |

Tabel 3.4.2.25 Use Case Scenario Hapus Komentar Member

**Tabel 3.4.2.26 Use Case Pulihkan Akun Member**

|  |  |
| --- | --- |
| CHORON-F-026 | Pulihkan Akun Member |
| Objective | Aktor ingin mengembalikan Akun member yang sebelumnya terhapus |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Member |
| Main flow | 1. Aktor melakukan klik Pulihkan pada salah satu akun member pada table “Telah Terhapus” 2. Sistem mengubah status member tersebut menjadi aktif di database |
| Alternative flow | - |
| Post-condition | User terdapat pada table User yang terlihat dan user dapat dilihat di website dan kembali menggunakan akunnya |

Tabel 3.4.2.26 Use Case Scenario Pulihkan Akun Member

**Tabel 3.4.2.27 Use Case Pulihkan Chord Member**

|  |  |
| --- | --- |
| CHORON-F-027 | Pulihkan Chord Member |
| Objective | Aktor ingin melakukan pemulihan terhadap chord yang sebelumnya terhapus |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Chord |
| Main flow | 1. Aktor melakukan klik pulihkan pada salah satu chord pada table “Telah Terhapus” 2. Sistem memindahkan status chord tersebut menjadi aktif |
| Alternative flow | - |
| Post-condition | Chord terdapat pada table Chord yang terlihat dan chord dapat dilihat di website |

Tabel 3.4.2.27 Use Case Scenario Pulihkan Chord Member

**Tabel 3.4.2.28 Use Case Pulihkan Komentar Member**

|  |  |
| --- | --- |
| CHORON-F-028 | Pulihkan Komentar Member |
| Objective | Aktor ingin melakukan pemulihan terhadap komentar yang sebelumnya terhapus |
| Actors | Admin |
| Pre-condition | Aktor telah melakukan login dan berada pada halaman Lihat Semua Komentar |
| Main flow | 1. Aktor melakukan klik pulihkan pada salah satu komentar pada table “Telah Terhapus” 2. Sistem memindahkan status komentar tersebut menjadi aktif |
| Alternative flow | - |
| Post-condition | Komentar terdapat pada table komentar yang terlihat dan komentar dapat dilihat di website |

Tabel 3.4.2.28 Use Case Scenario Pulihkan Chord Member

### Sequence Diagram

#### LOGIN (CHORON-F-001)

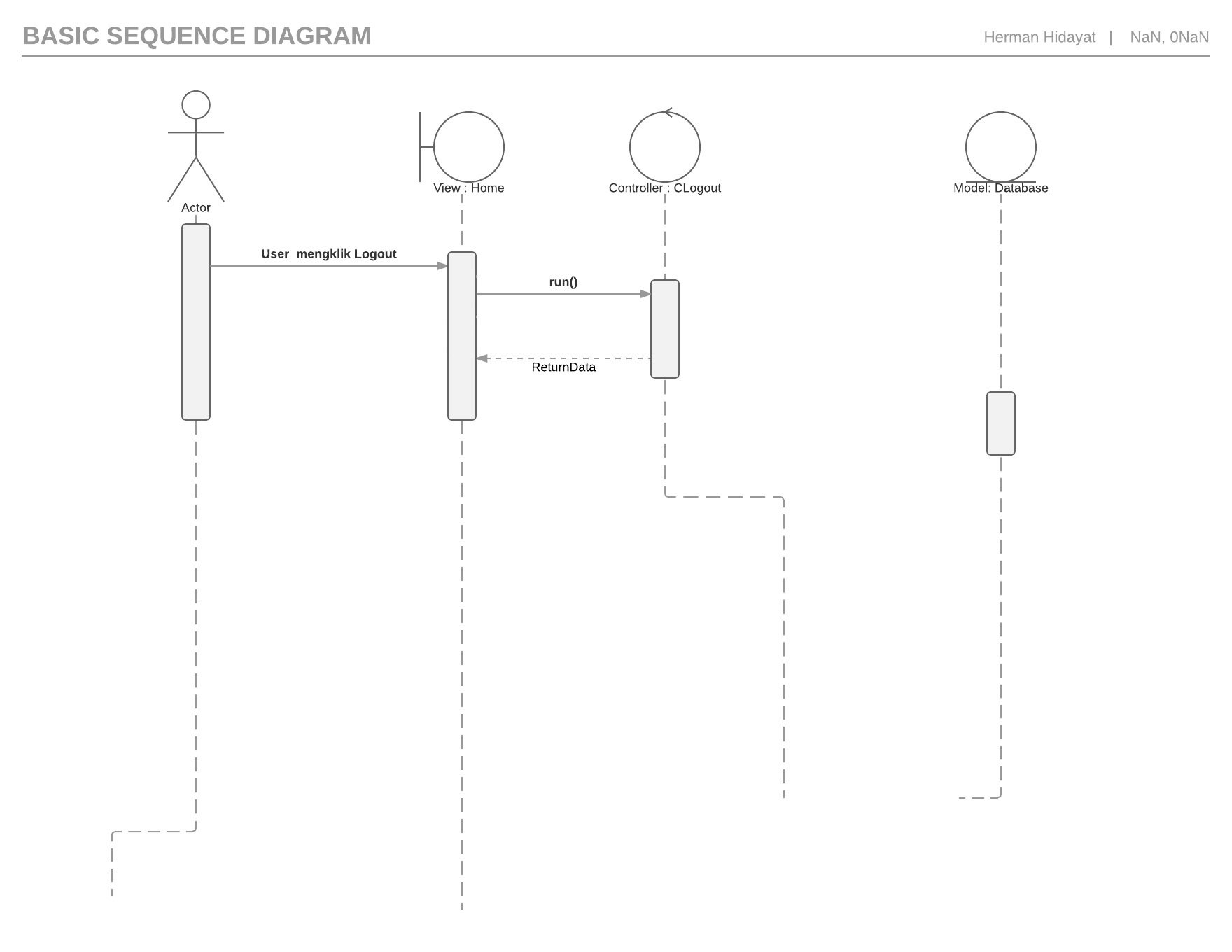
Untuk login, user harus mengisi form login terlebih dahulu. Sistem akan membandingkan inputan user dengan data di database. Apabila username dan password valid, user akan diredirect menuju halaman utama, sedangkan apabila username dan password tidak valid, user akan diredirect kembali menuju form login.



#### Diagram 3.4.3.1 Login

#### 3.4.3.2 Logout (CHORON-F-002)

1. Setelah login, user dapat melakukan logout. Agar user dapat logout, Controller CLogout akan menjalankan method run dan meredirect user menuju halaman utama



#### Diagram 3.4.3.2 Logout

#### Registrasi Akun (CHORON-F-003)

Untuk registrasi, Guest harus merequest Form Registrasi terlebih dahulu. Setelah data diinputkan dan guest mengklik daftar, system akan memeriksa apakah data yang dimasukkan sudah valid. Jika datanya valid system akan menyimpannya di database lalu meredirect menuju halaman login, sedangkan apabila inputan tidak valid, Guest akan diredirect menuju halaman FormRegistrasi

.

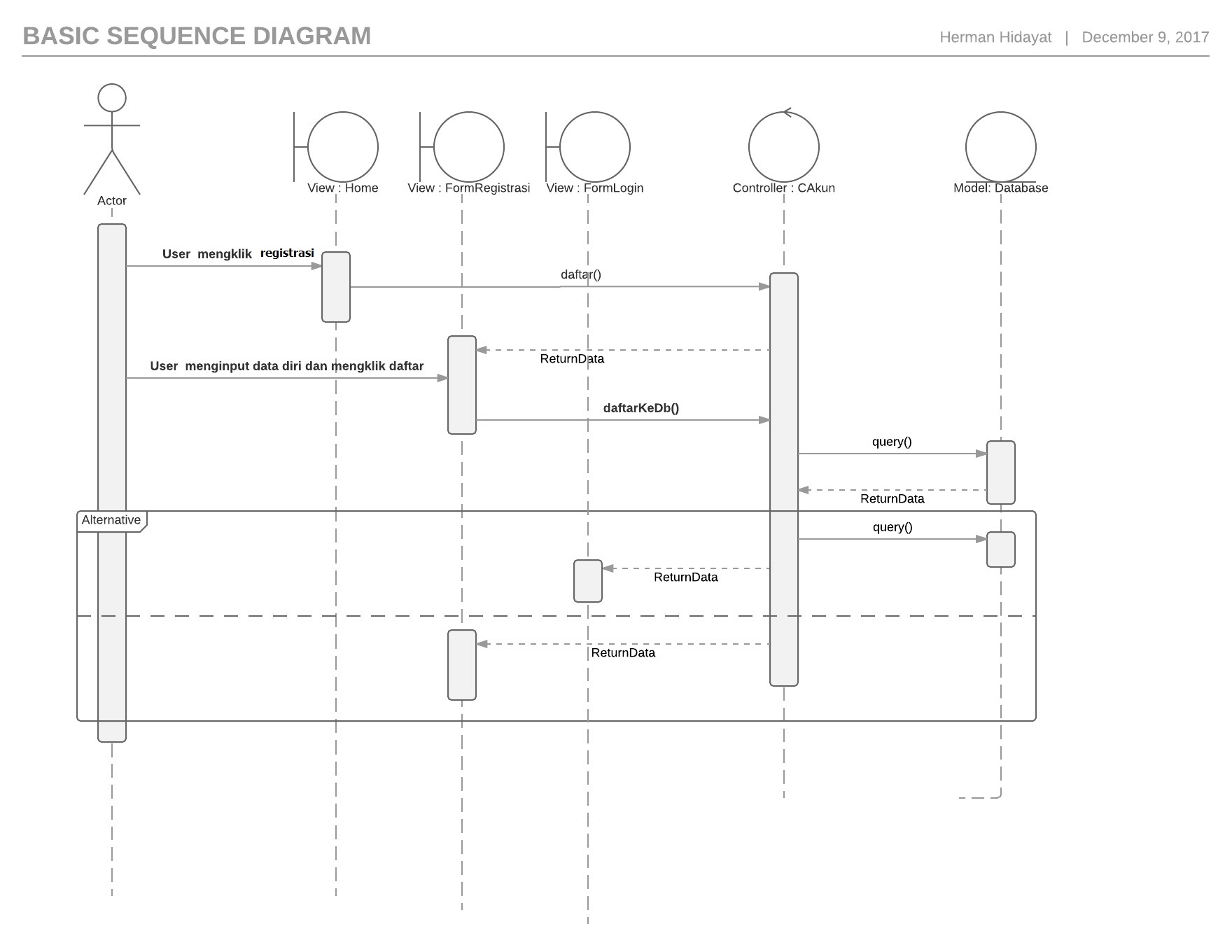
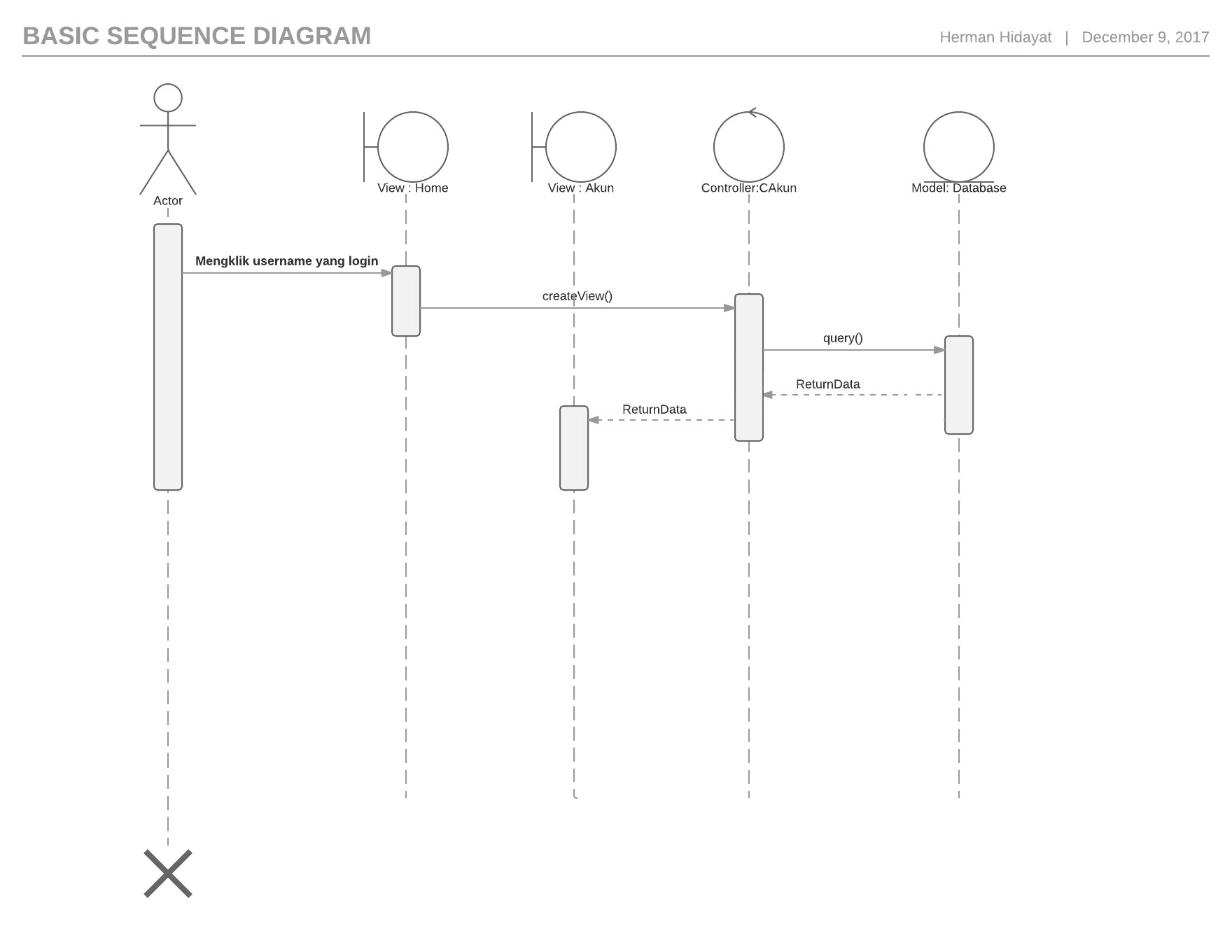


Diagram 3.4.3.3 Registrasi Akun

#### Lihat Akun (CHORON-F-004)

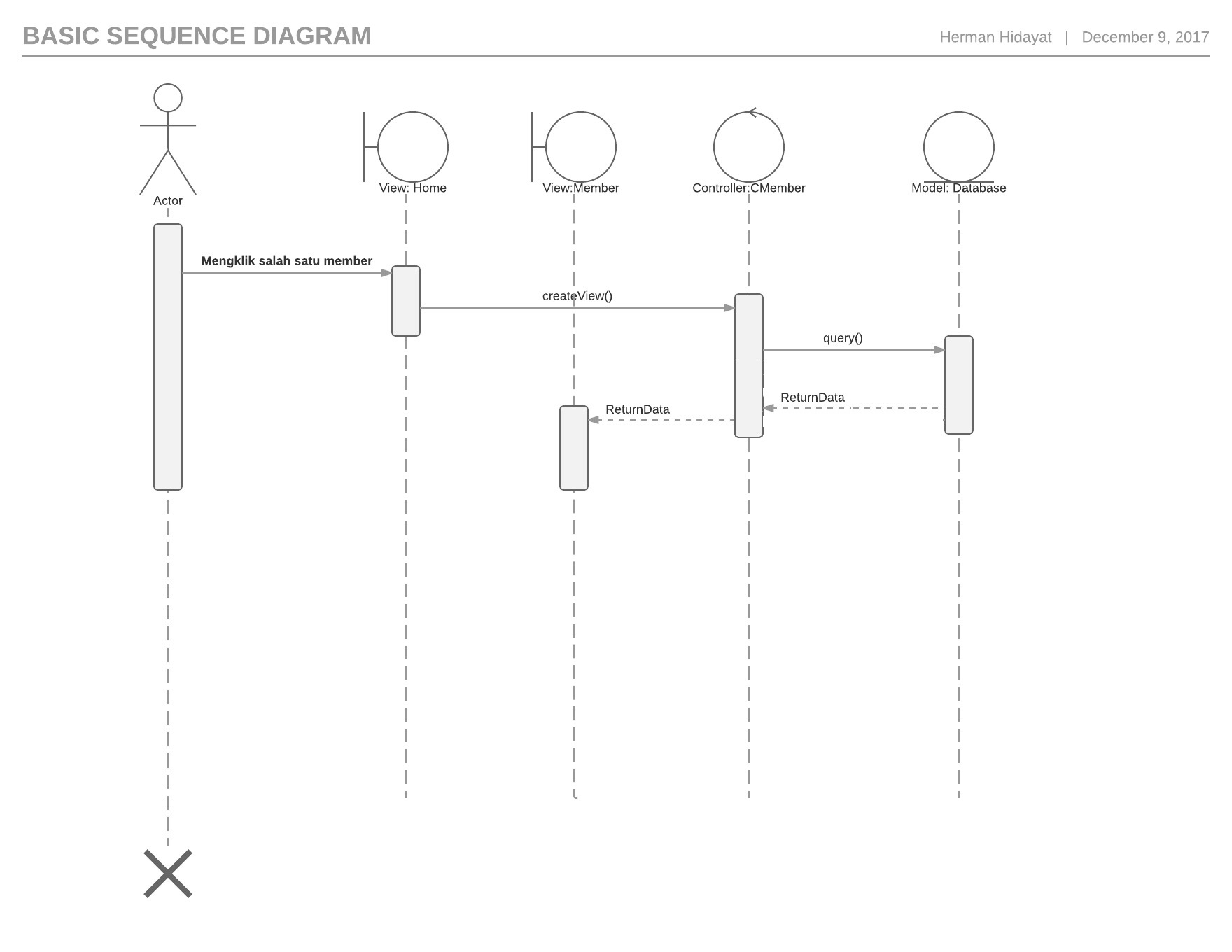
Agar member user dapat melihat akunnya, member harus login terlebih dahulu, kemudian mengklik usernamenya yang sedang login. Controller CAkun akan menjalankan fungsi create viewnya, mengambil data dari database dan menampilkannya kepada member dengan view Akun.



#### Diagram 3.4.3.4 Lihat Akun

#### Lihat Member (CHORON-F-005)

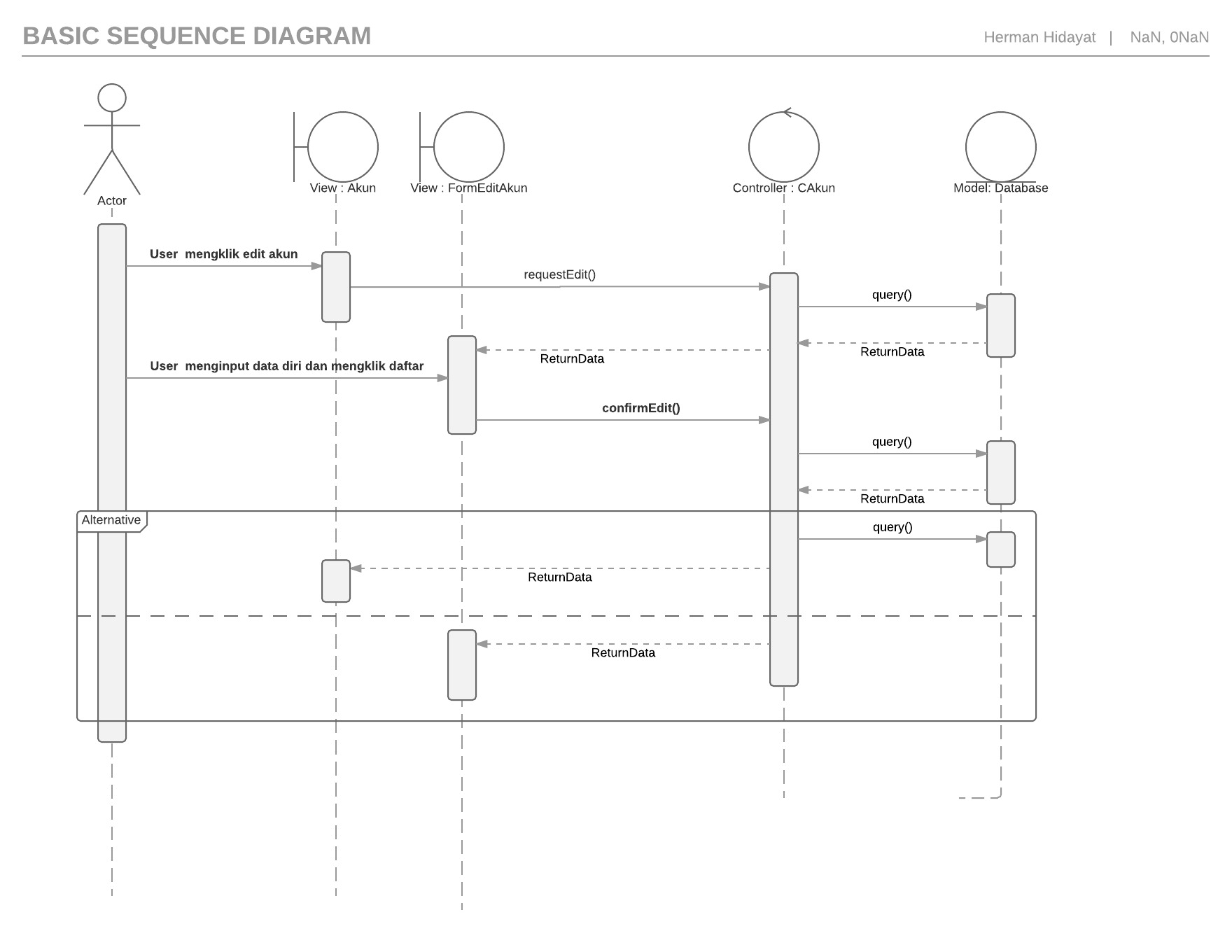
Untuk melihat profil member lain yang sudah terdaftar, user harus mengklik salah satu username member. Controller CMember akan menjalankan fungsi createViewnya, mengambil data dari database, dan menampilkan profil member yang dipilih lewat view Member.



#### Diagram 3.4.3.5 lihat member

#### Edit Akun (CHORON-F-006)

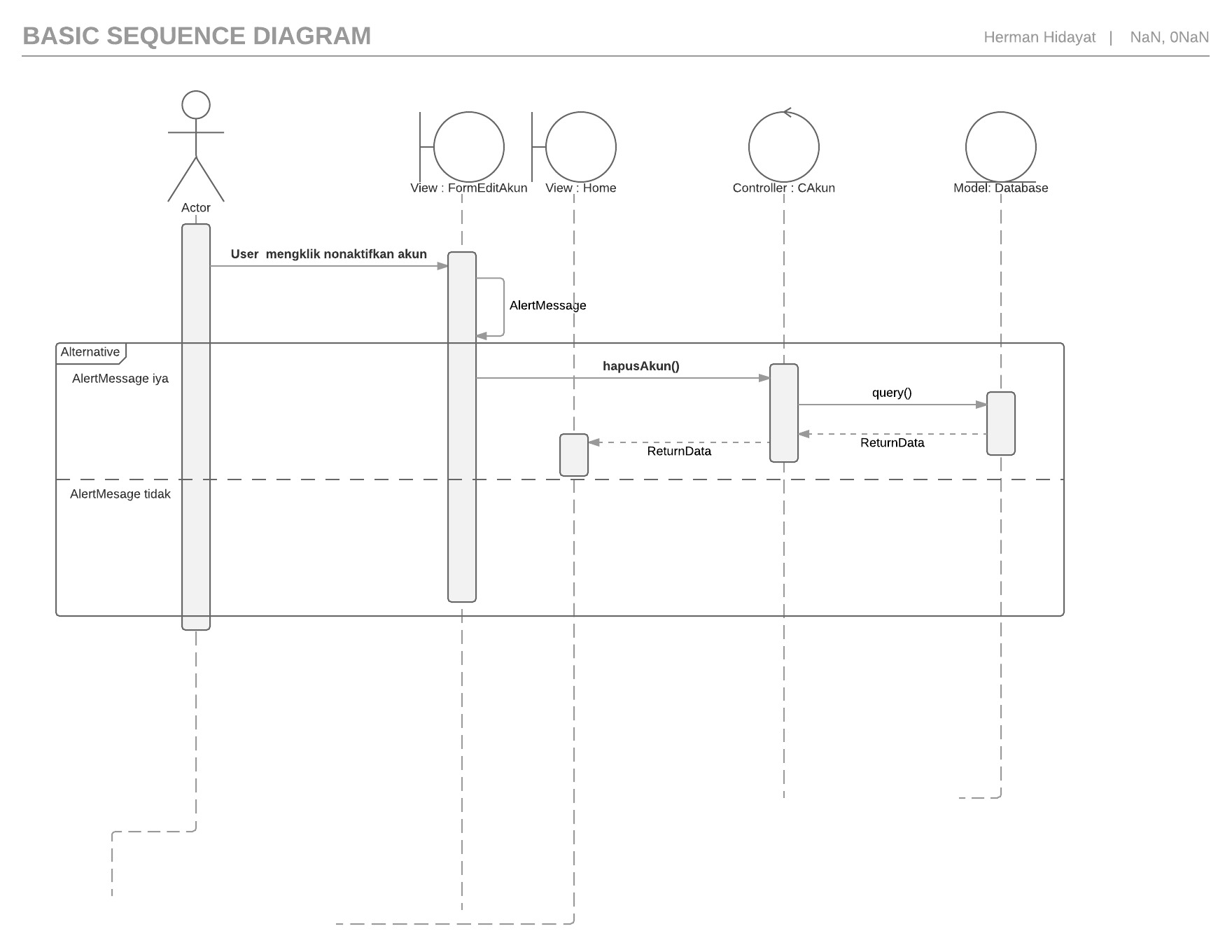
Untuk mengubah profil data dirinya, member harus merequest lihat akun terlebih dahulu, kemudian mengklik edit akun. Controller CAkun akan menjalankan fungsi editAkunRequest() yaitu mengambil data akun saat ini, dan menyajikannya lewat view FormEditAkun(). Pada view ini user dapat mengubah profilnya dan mengklik simpan. Kemudian Controller CAkun akan menjalankan konfirmasiEdit() dimana apabila profil yang baru tidak valid, member akan dikembalikan ke halaman FormEditAkun dan apabila valid data akan disimpan dan user akan diredirect menuju halaman Akun



#### Diagram 3.4.3.6 Edit Akun

#### Nonaktifkan Akun (CHORON-F-007)

Untuk menonaktifkan akun, member harus merequest edit akun terlebih dahulu. Ketika member mengklik nonaktifkan akun, akan muncul peringatan. Jika user mengklik iya, system akan menjalankan fungsi hapus akun, mengubah databasenya, dan meredirect user menuju halaman home. Sedangkan jika user memilih tidak, user tetap dapat mengedit akunnya.



#### Diagram 3.4.3.7 Nonaktifkan Akun

#### Lihat Komentar (CHORON-F-008)

Untuk membuat chord, member harus login terlebih dahulu. Ketika member mengklik submit chord, Controller CChord akan menampilkan view FormBuatChord dengan menjalankan fungsi buatChord(). Kemudian member mengisi form chord dan kemudian mengklik submit. Untuk menyimpan chord yang disubmit user, Controller CChord akan menjalankan fungsi inputChordKeDb() dan kemudian meredirect user menuju halaman home

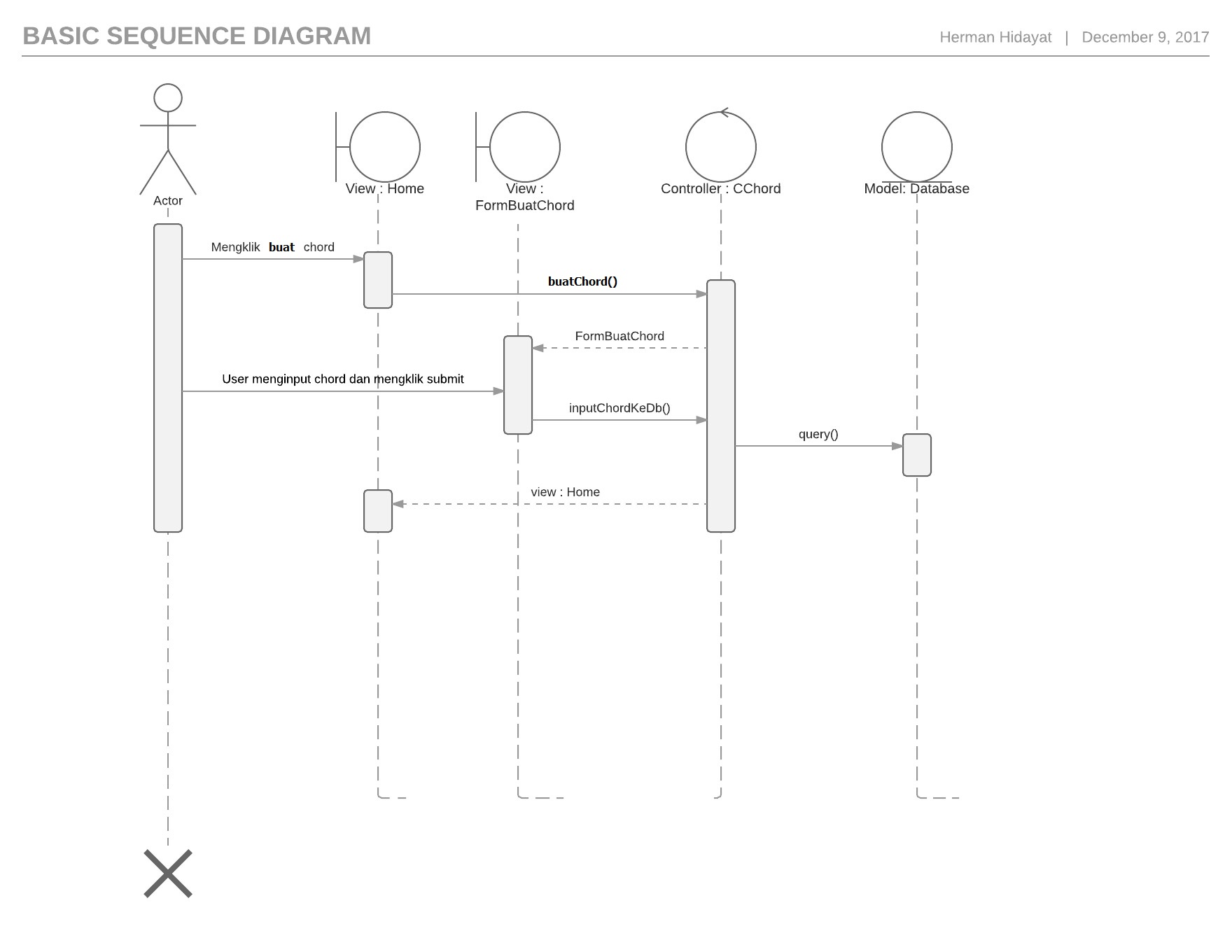


Diagram 3.4.3.8 Buat Chord

#### Cari Kata Kunci (CHORON-F-009)

Untuk melihat chord, user harus mengklik salah satu chord. Controller CChord akan menjalankan fungsi createViewnya, mengambil data dari database, dan menampilkan chord member yang dipilih lewat view Chord.

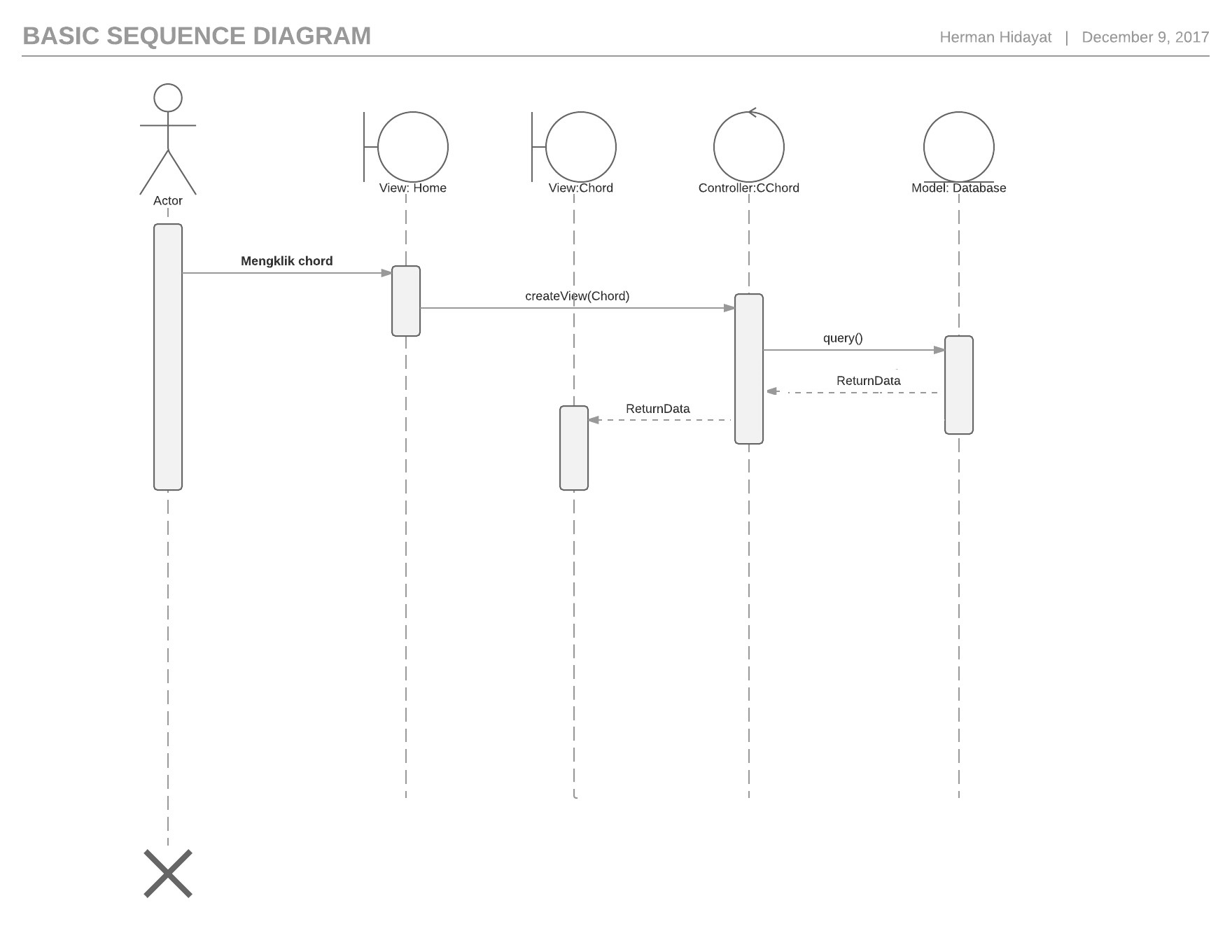
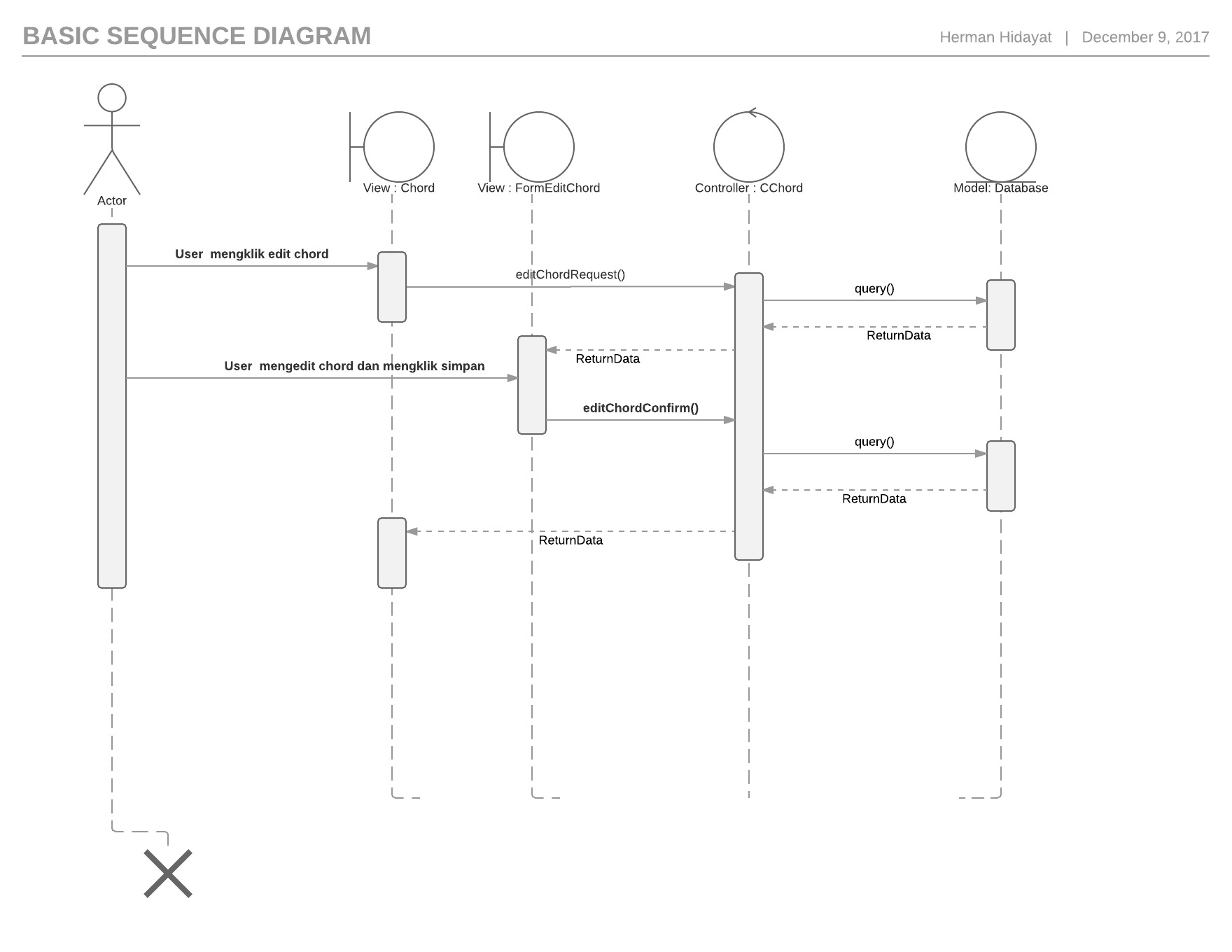


Diagram 3.4.3.9 Lihat Chord

#### Edit Chord (CHORON-F-010)

Untuk mengubah chord mereka, member harus merequest lihat chord terlebih dahulu, kemudian mengklik edit chord. Controller CChord akan menjalankan fungsi editChordRequest() yaitu mengambil data chord saat ini, dan menyajikannya lewat view FormEditChord(). Pada view ini user dapat mengubah chord dan mengklik simpan. Kemudian Controller CChord akan menjalankan editChordConfirm() dimana apabila chord yang baru tidak valid, member akan dikembalikan ke halaman FormEditChord dan apabila valid data akan disimpan dan user akan diredirect menuju halaman Chord



#### Diagram 3.4.3.10 Edit Chord

#### 3.4.3.11 Hapus Chord (CHORON-F-011)

Untuk menghapus chord, member harus merequest edit chord terlebih dahulu pada chord yang diinginkan. Ketika member mengklik hapus chord, akan muncul peringatan. Jika user mengklik iya, system akan menjalankan fungsi hapusChord, mengubah databasenya, dan meredirect user menuju halaman akun. Sedangkan jika user memilih tidak, user tetap dapat mengedit chordnya.

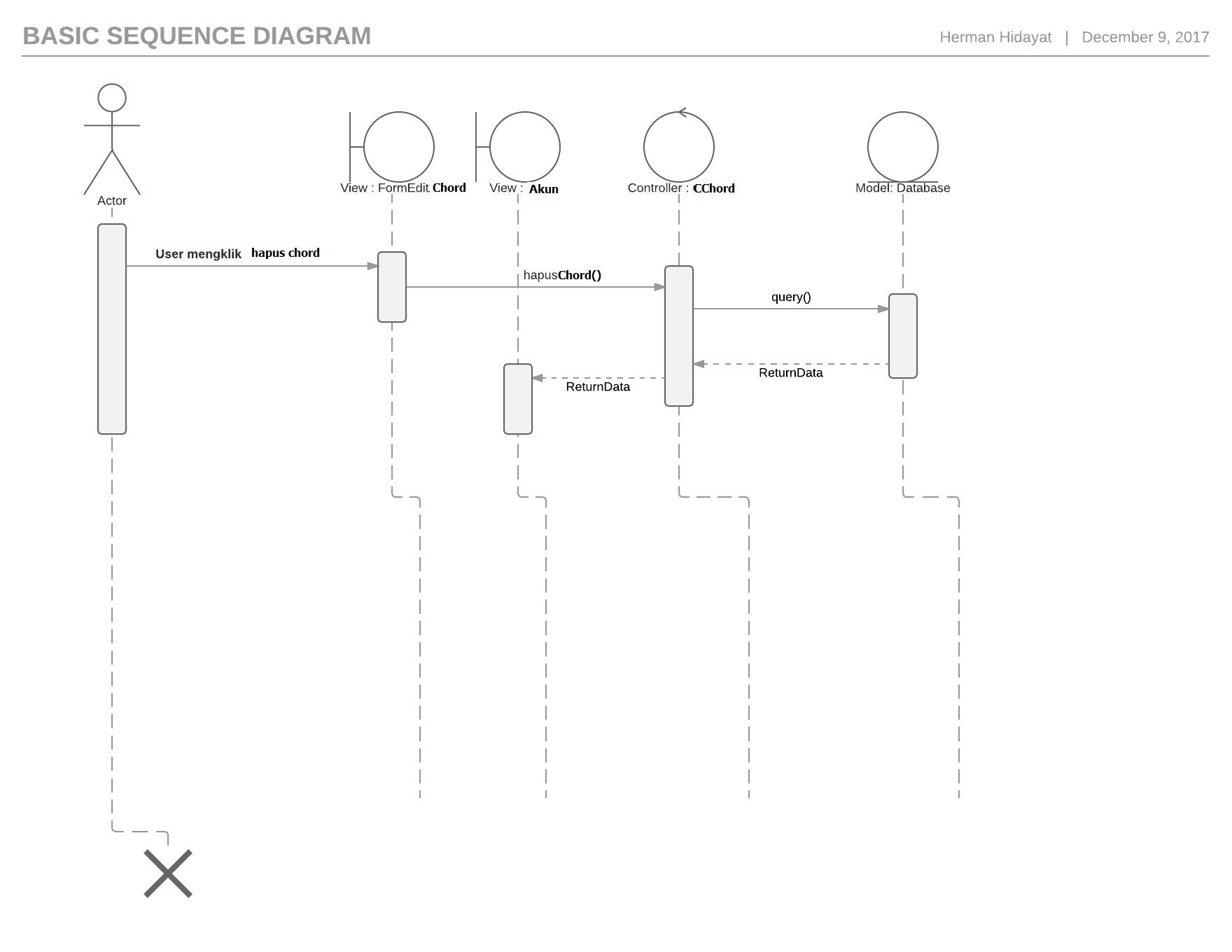
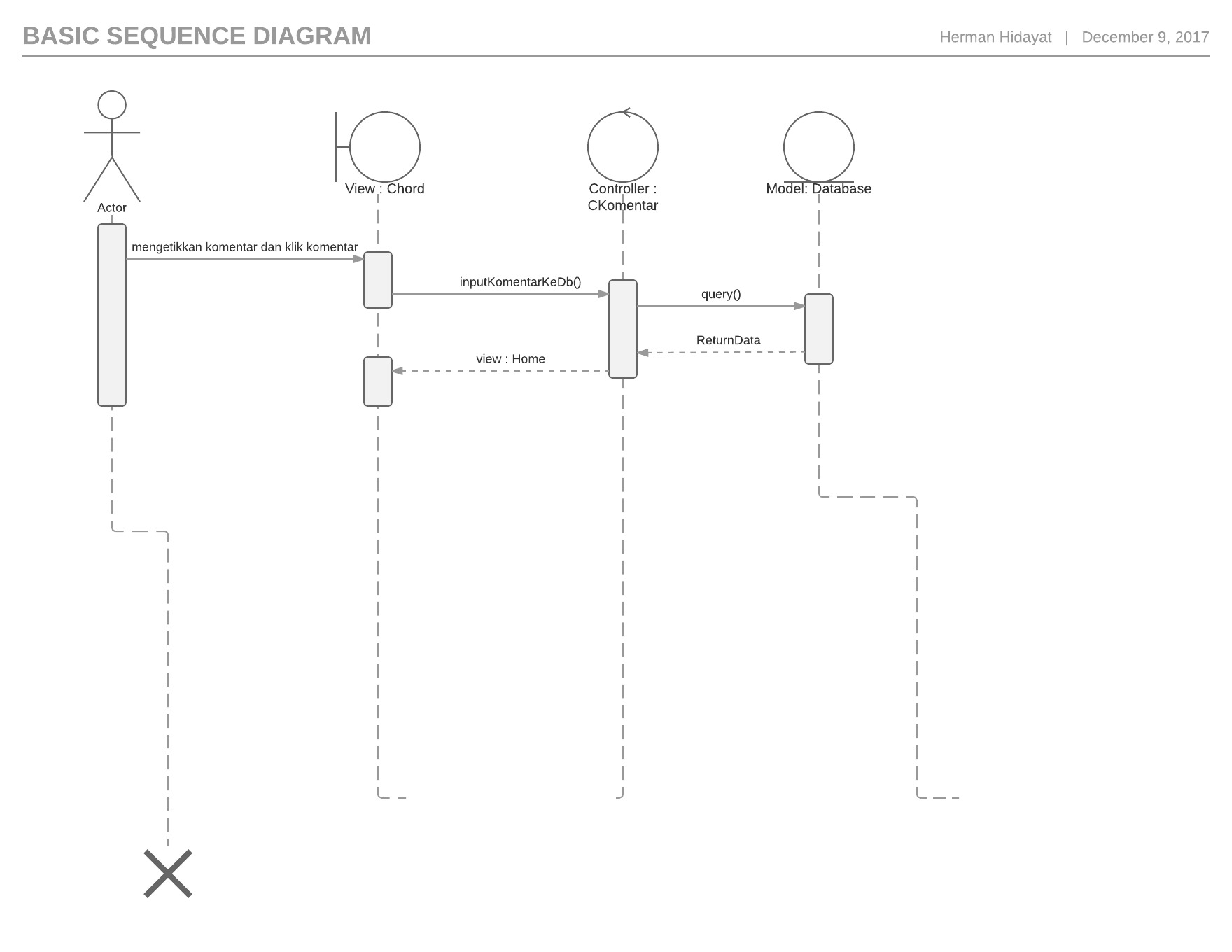


Diagram 3.4.3.11 Hapus Chord

#### 3.4.3.12 Beri Komentar (CHORON-F-012)

Untuk Beri Komentar, Member perlu mengetik dan klik komentar pada view Chord. Controller CKomentar akan menjalakan method inputKomentarKeDb() yang nantinya data tersebut akan dimasukkan ke database oleh query(). Setelah itu data dikembalikkan ke Controller CKomentar yang nantinya akan menampilkan view Chord.

  
Diagram 3.4.3.12 Beri Komentar

#### 3.4.3.13 Edit Komentar (CHORON-F-013)

Untuk mengedit komentar, member harus mengeklik edit komentar terlebih dahulu. Controller CKomentar akan menjalankan fungsi editKomentarRequest() yaitu mengambil data chord saat ini, dan menyajikannya lewat view FormEditKomentar(). Pada view ini user dapat mengubah komentar dan mengklik simpan. Kemudian Controller CKomentar akan menjalankan editKomentarConfirm() dimana apabila chord yang baru tidak valid, member akan dikembalikan ke halaman FormEditKomentar dan apabila valid data akan disimpan dan user akan diredirect menuju halaman Chord.

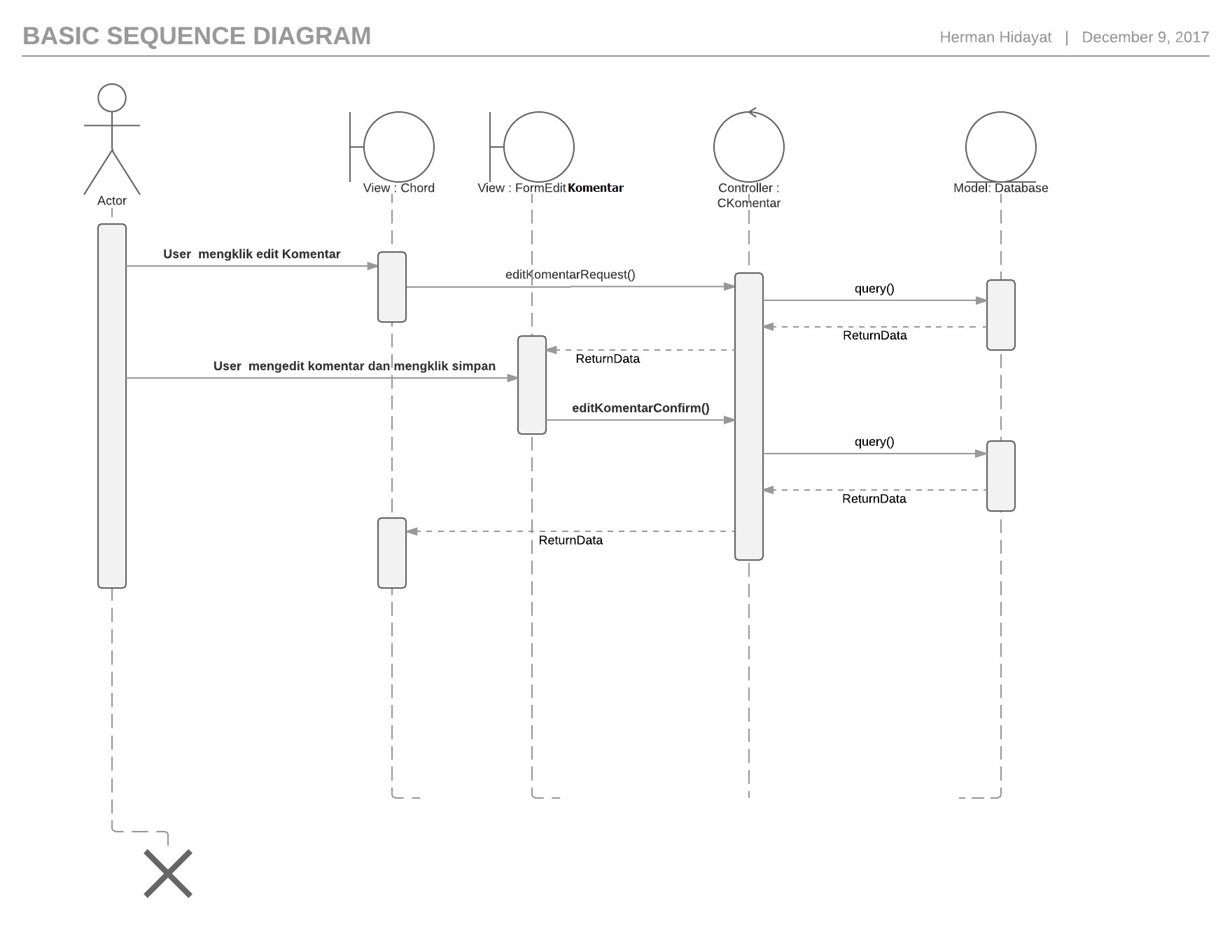


Diagram 3.4.3.13 Edit Komentar

#### 3.4.3.14 Hapus Komentar (CHORON-F-014)

Untuk menghapus komentar, member harus login dan merequest edit komentar terlebih dahulu, kemudian controller CKomentar akan menjalankan method hapus komentar, melakukan perubahan pada database dan kemudian meredirect member ke halaman chord

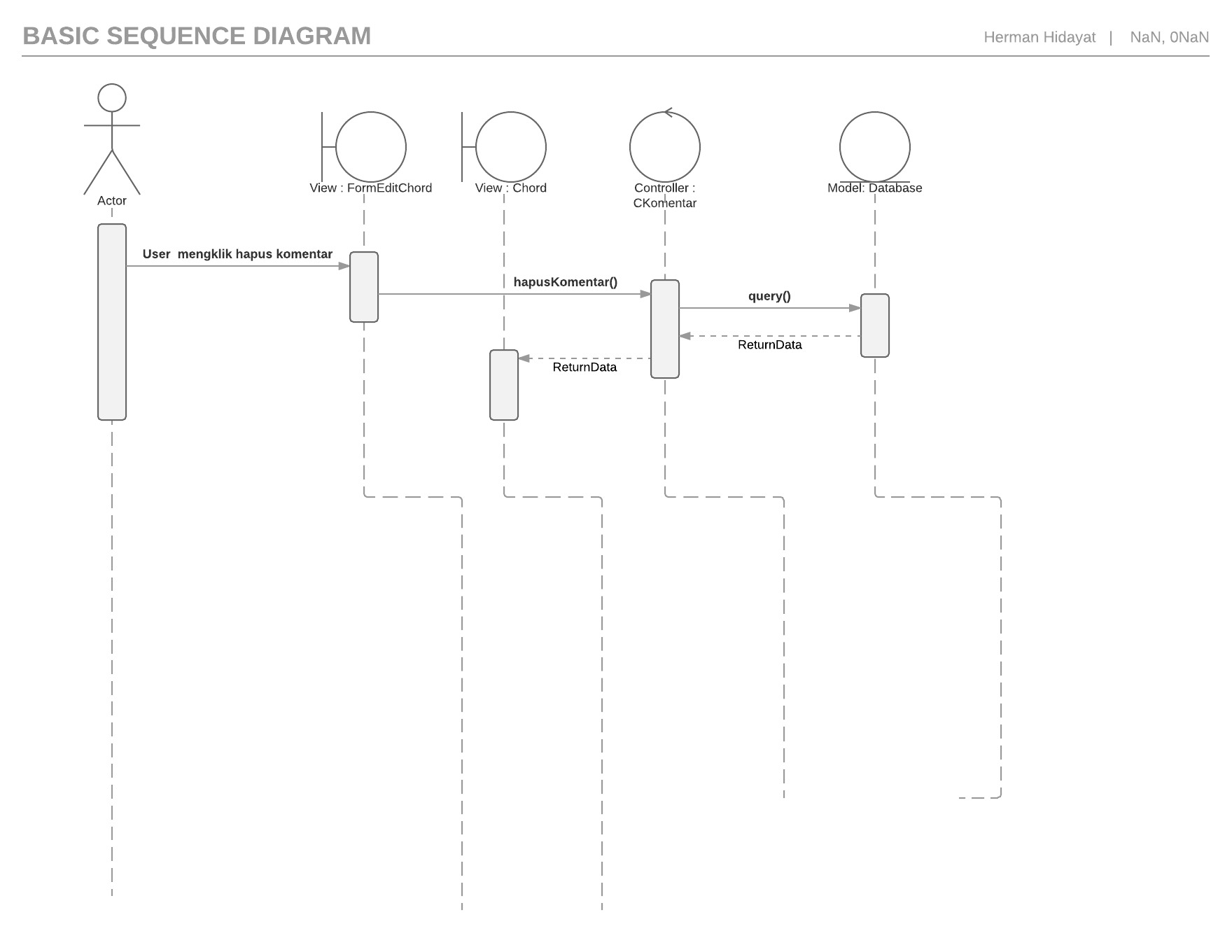


Diagram 3.4.3.15 Hapus Komentar

#### 3.4.3.15 Beri Rate (CHORON-F-015)

Untuk memberi rating, member harus memilih rating dan mengeklik rate terlebih dahulu pada view Chord. Controller CChord akan menjalankan fungsi beriRate() yaitu mengambil data rating yang terpilih. Lalu Controller CChord akan menjalankan tambahRate() jika member belum melakukan rate pada chord tersebut. Jika sebelumnya member sudah melakukan rate, maka fungsi ubahRate() akan dijalankan. Jika fungsi yang dibutuhkan telah berjalan, member akan didirect ke view Chord tersebut.

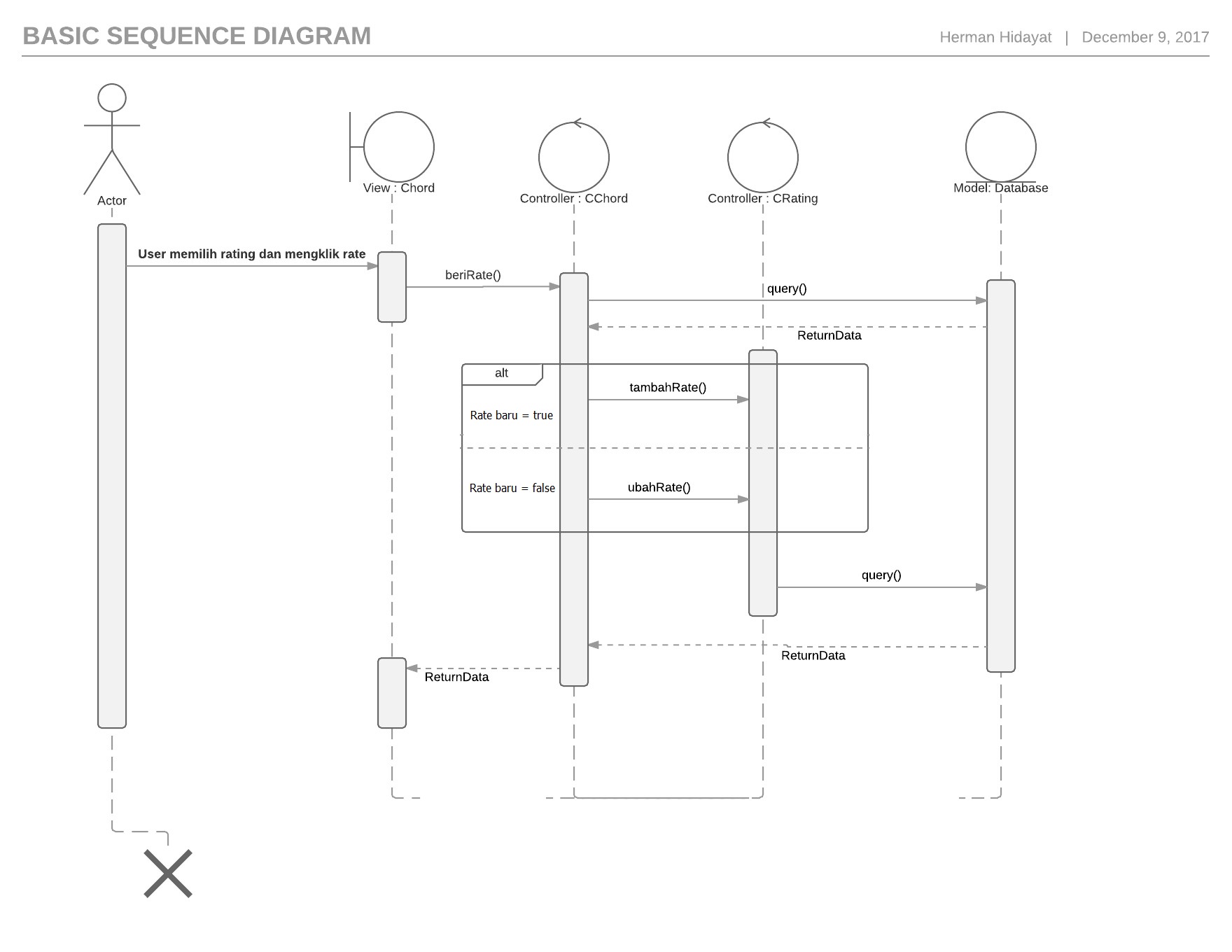


Diagram 3.4.3.15 Beri Rate

#### 3.4.3.16 Transpose (CHORON-F-016)

Untuk melakukan tranpose, member harus memilih step tranpose dan mengeklik tranpose terlebih dahulu pada view Chord. Controller Tranpose akan menjalankan fungsi tranpose() yang nantinya akan dikembalikan ke view Chord untuk menampilkan yang telah ditranpose tersebut.

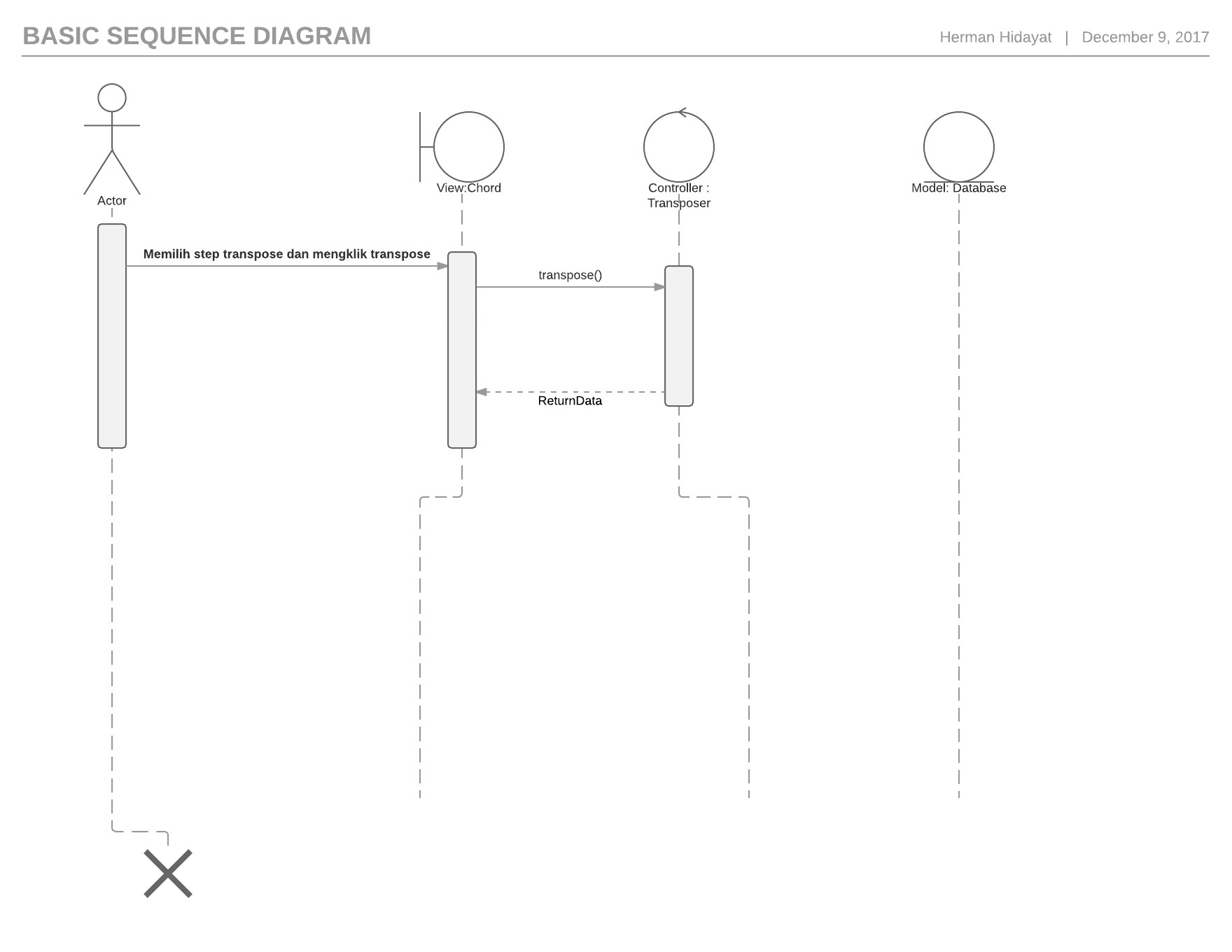


Diagram 3.4.3.16 Transpose

#### Melihat Semua Akun Member (CHORON-F-017)

Untuk melihat akun semua member, admin harus mengklik lihat semua member pada view AdminDashboard. Controller CAdmin akan menjalankan fungsi lihatSemuaMember() yang nantinya data dari database akan dikembalikan ke view AdminSemuaMember.

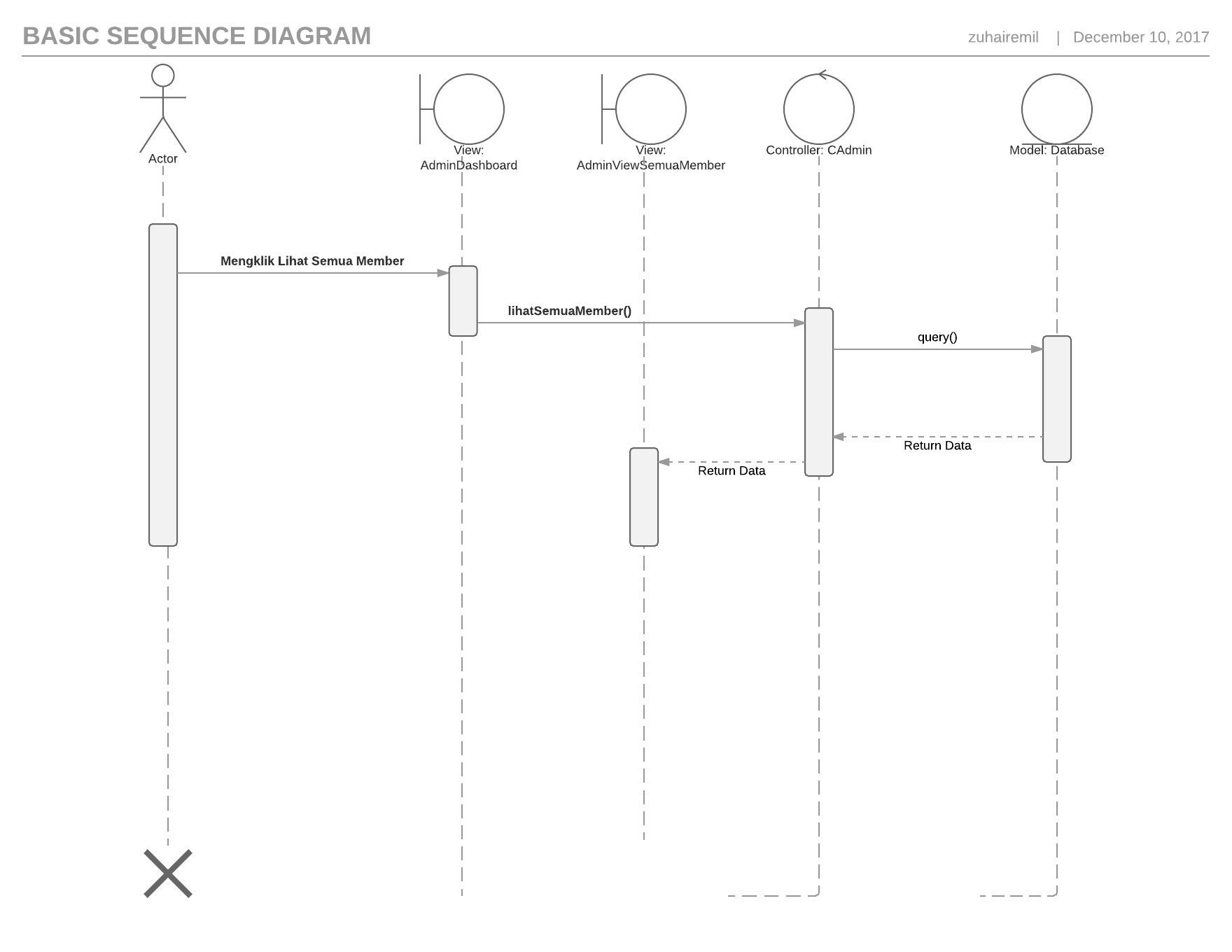


Diagram 3.4.3.17 Melihat Semua Akun Member

#### Melihat Semua Chord Member (CHORON-F-018)

Untuk melihat chord semua member, admin harus mengklik lihat semua chord pada view AdminDashboard. Controller CAdmin akan menjalankan fungsi lihatSemuaMember() yang nantinya data dari database akan dikembalikan ke view AdminSemuaChord.



Diagram 3.4.3.18 Melihat Semua Chord Member

#### Melihat Semua Komentar Member (CHORON-F-019)

Untuk melihat komentar semua member, admin harus mengklik lihat semua komentar pada view AdminDashboard. Controller CAdmin akan menjalankan fungsi lihatSemuaKomentar() yang nantinya data dari database akan dikembalikan ke view AdminSemuaKomentar.

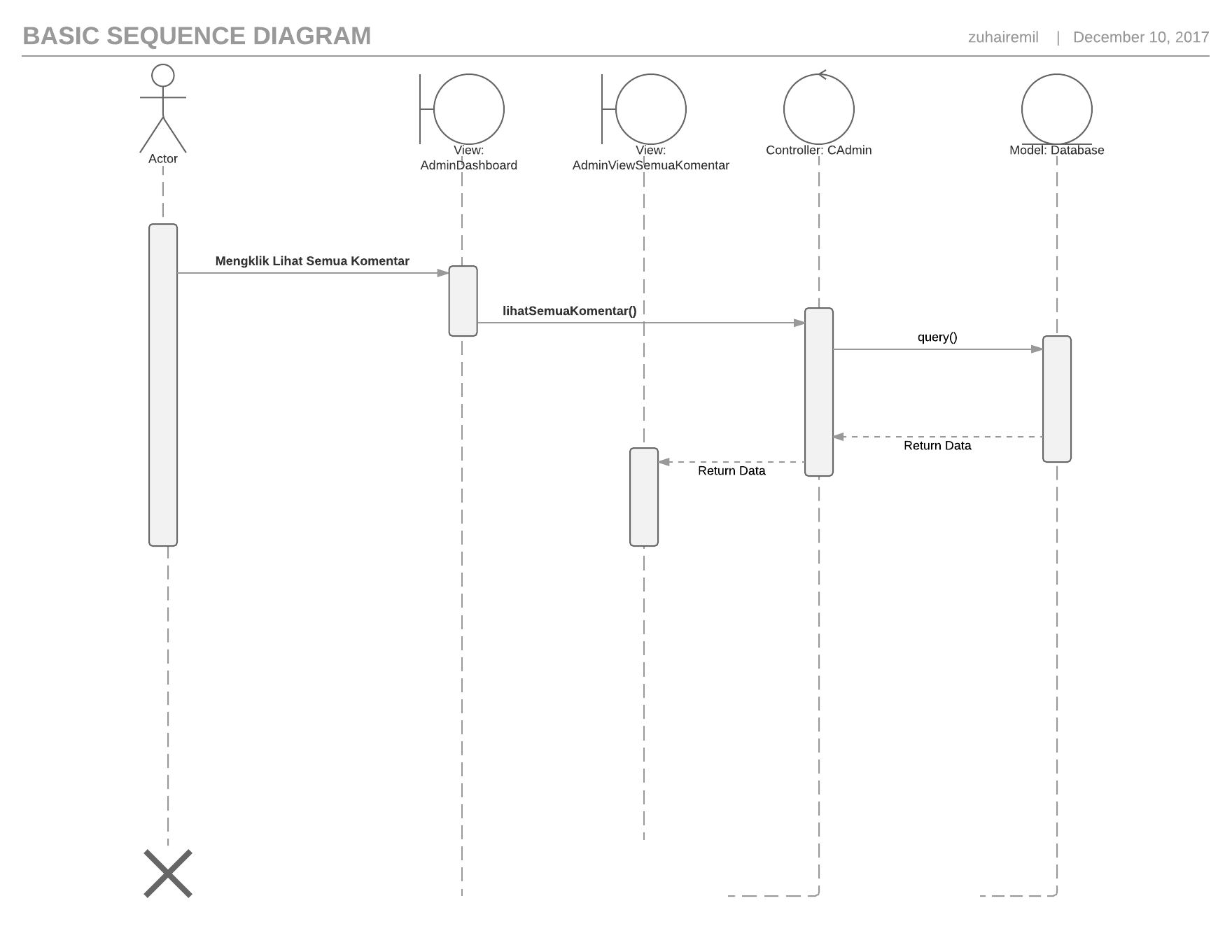


Diagram 3.4.3.19 Melihat Semua Komentar Member

#### Edit Akun Member (CHORON-F-020)

Untuk mengedit akun member, admin harus mengklik edit di salah satu member pada view AdminViewSemuaMember. Controller CAdminSemuaMember akan menjalankan fungsi requestEdit() yang nantinya akan menampilkan view AdminFormEditMember dengan nilai dari database yang sesuai. Setelah tombol konfirmasi diklik, CAdminSemuaMember menjalankan fungsi konfirmasiEdit(). Jika input yang dimasukkan valid, maka akan kembali ke view AdminSemuaMember. Jika input yang dimasukkan tidak valid, maka akan dikembalikkan ke view AdminFormEditMember.

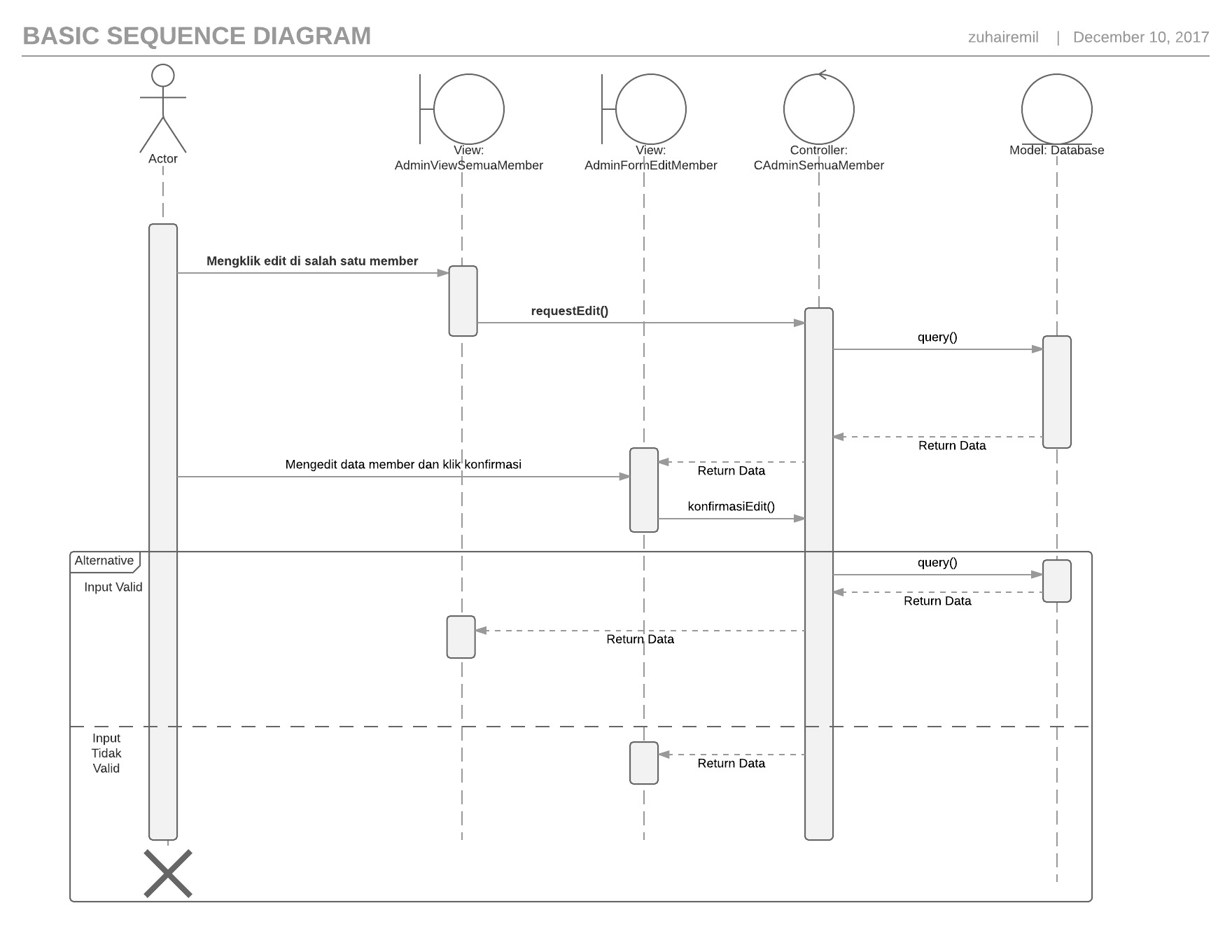


Diagram 3.4.3.20 Edit Akun Member

#### Edit Chord Member (CHORON-F-021)

Untuk mengedit chord member, admin harus mengklik edit di salah satu chord pada view AdminViewSemuaChord. Controller CAdminSemuaChord akan menjalankan fungsi editChordRequest () yang nantinya akan menampilkan view AdminFormEditChord dengan nilai dari database yang sesuai. Setelah tombol konfirmasi diklik, CAdminSemuaChord menjalankan fungsi editChordConfirm(). Jika input yang dimasukkan valid, maka akan kembali ke view AdminSemuaChord. Jika input yang dimasukkan tidak valid, maka akan dikembalikkan ke view AdminFormEditChord.

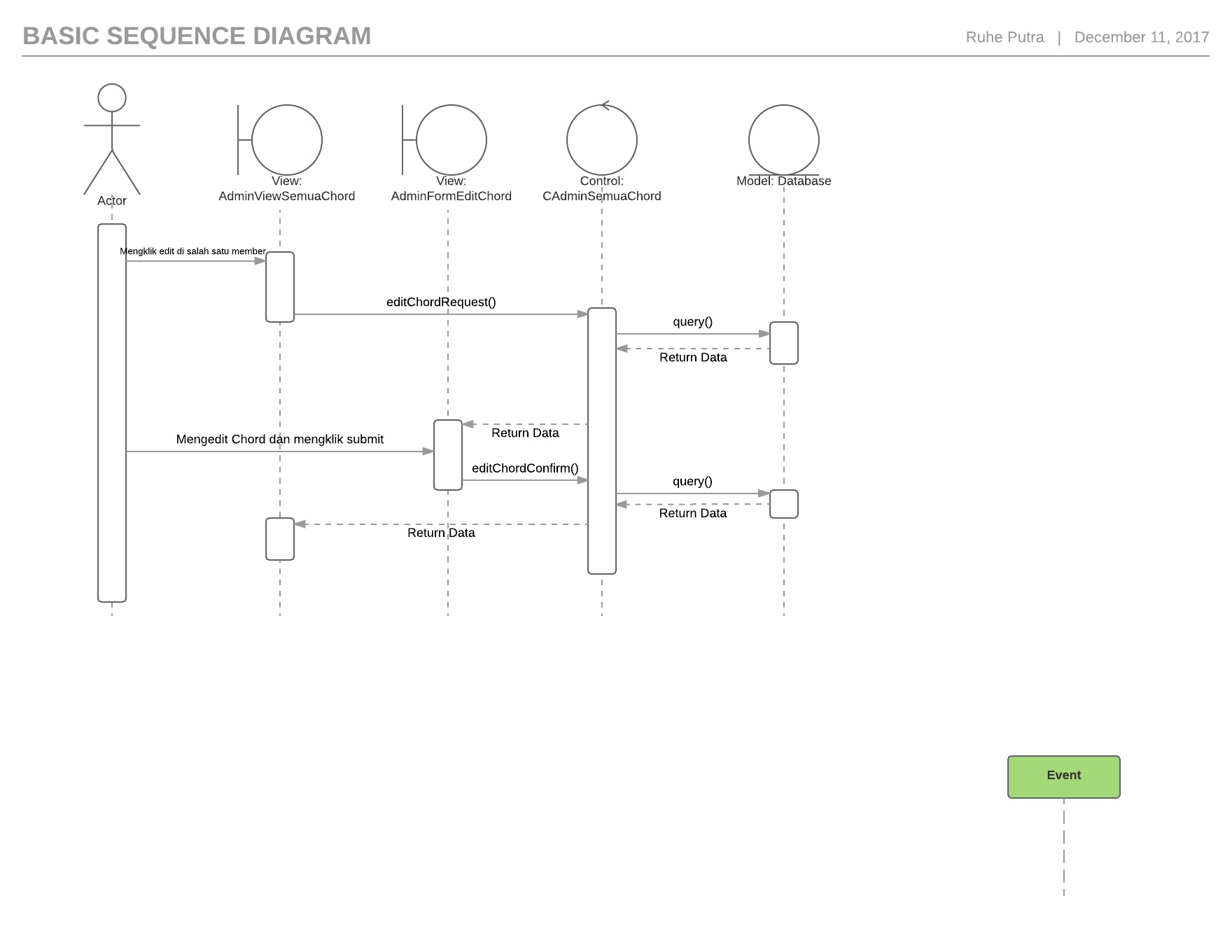


Diagram 3.4.3.21 Edit Chord Member

#### Edit Komentar Member (CHORON-F-022)

Untuk mengedit komentar member, admin harus mengklik edit di salah satu komentar pada view AdminViewSemuaKomentar. Controller CAdminSemuaKomentar akan menjalankan fungsi editKomentarRequest () yang nantinya akan menampilkan view AdminFormEditKomentar dengan nilai dari database yang sesuai. Setelah tombol konfirmasi diklik, CAdminSemuaKomentar menjalankan fungsi editKomentarConfirm(). Jika input yang dimasukkan valid, maka akan kembali ke view AdminSemuaKomentar. Jika input yang dimasukkan tidak valid, maka akan dikembalikkan ke view AdminFormEditKomentar.

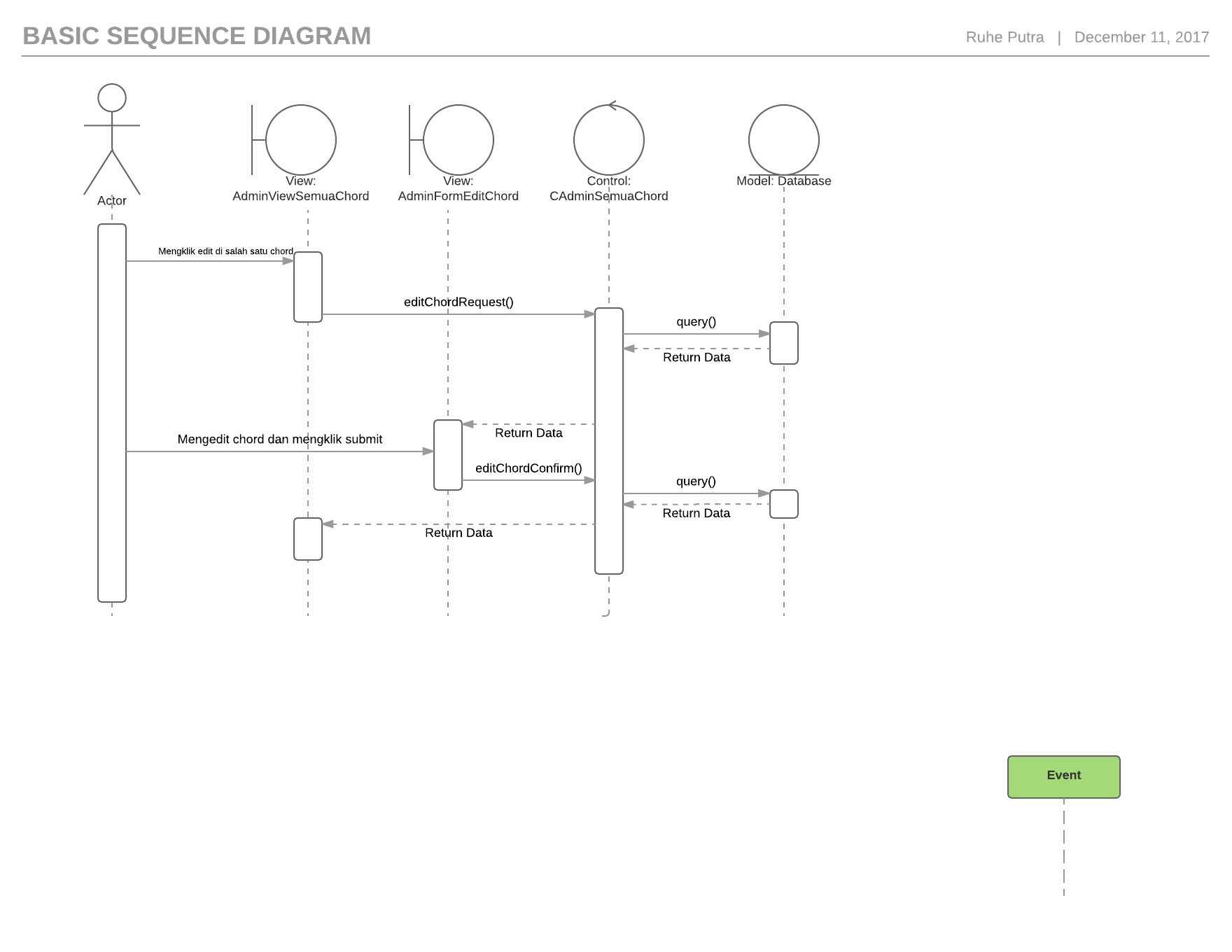


Diagram 3.4.3.22 Edit Komentar Member

#### Hapus Akun Member (CHORON-F-023)

Untuk menghapus akun member, admin harus mengklik delete di salah satu member pada view AdminViewSemuaMember. Controller CAdminSemuaMember akan menjalankan fungsi hapusAkun() yaitu menjalankan query untuk menghapus akun member.



Diagram 3.4.3.23 Hapus Akun Member

#### Hapus Chord Member (CHORON-F-024)

Untuk menghapus chord member, admin harus mengklik delete di salah satu chord pada view AdminViewSemuaChord. Controller CAdminSemuaChord akan menjalankan fungsi hapusChord() yaitu menjalankan query untuk menghapus chord member.

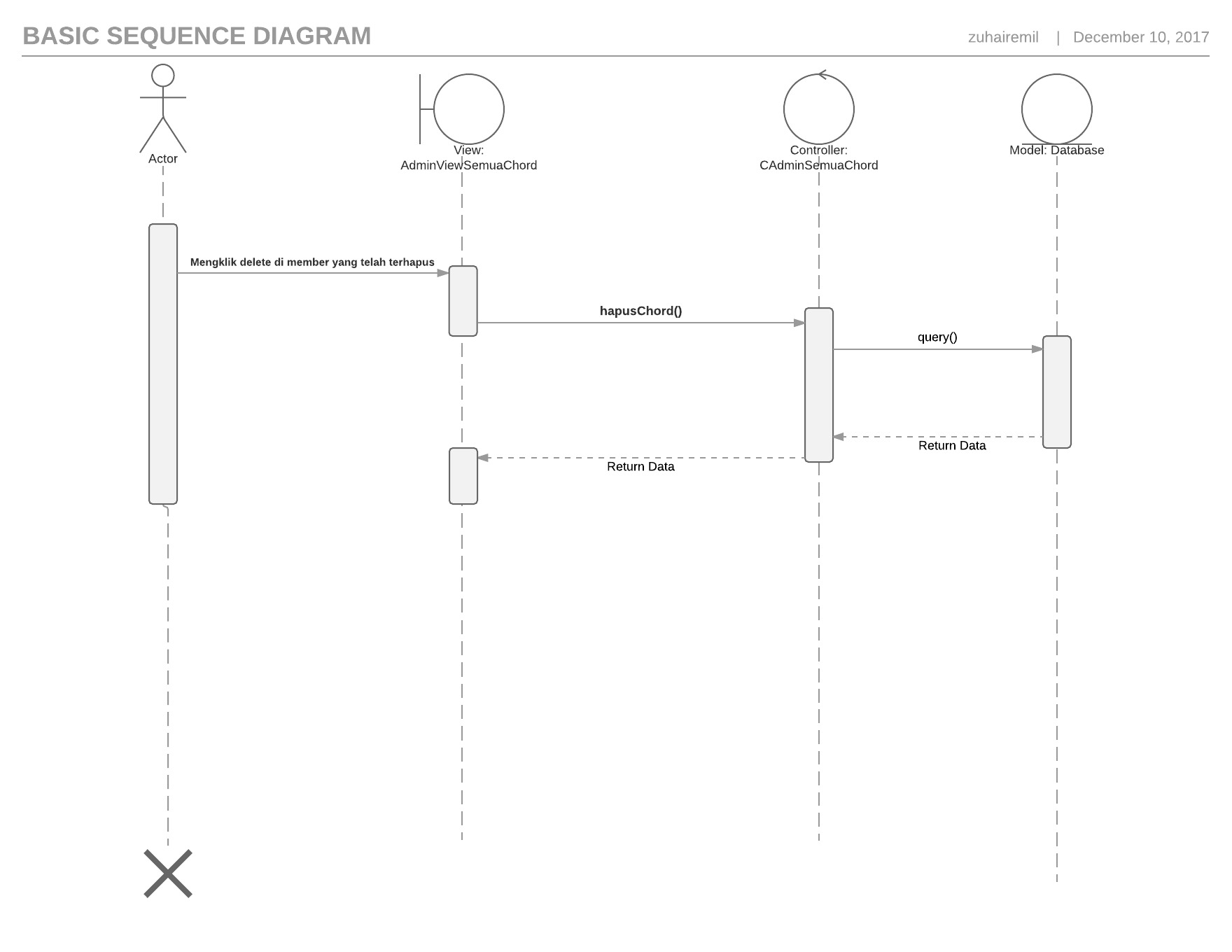


Diagram 3.4.3.24 Hapus Chord Member

#### Hapus Komentar Member (CHORON-F-025)

Untuk menghapus komentar member, admin harus mengklik delete di salah satu komentar pada view AdminViewSemuaKomentar. Controller CAdminSemuaKomentar akan menjalankan fungsi hapusKomentar() yaitu menjalankan query untuk menghapus komentar member.

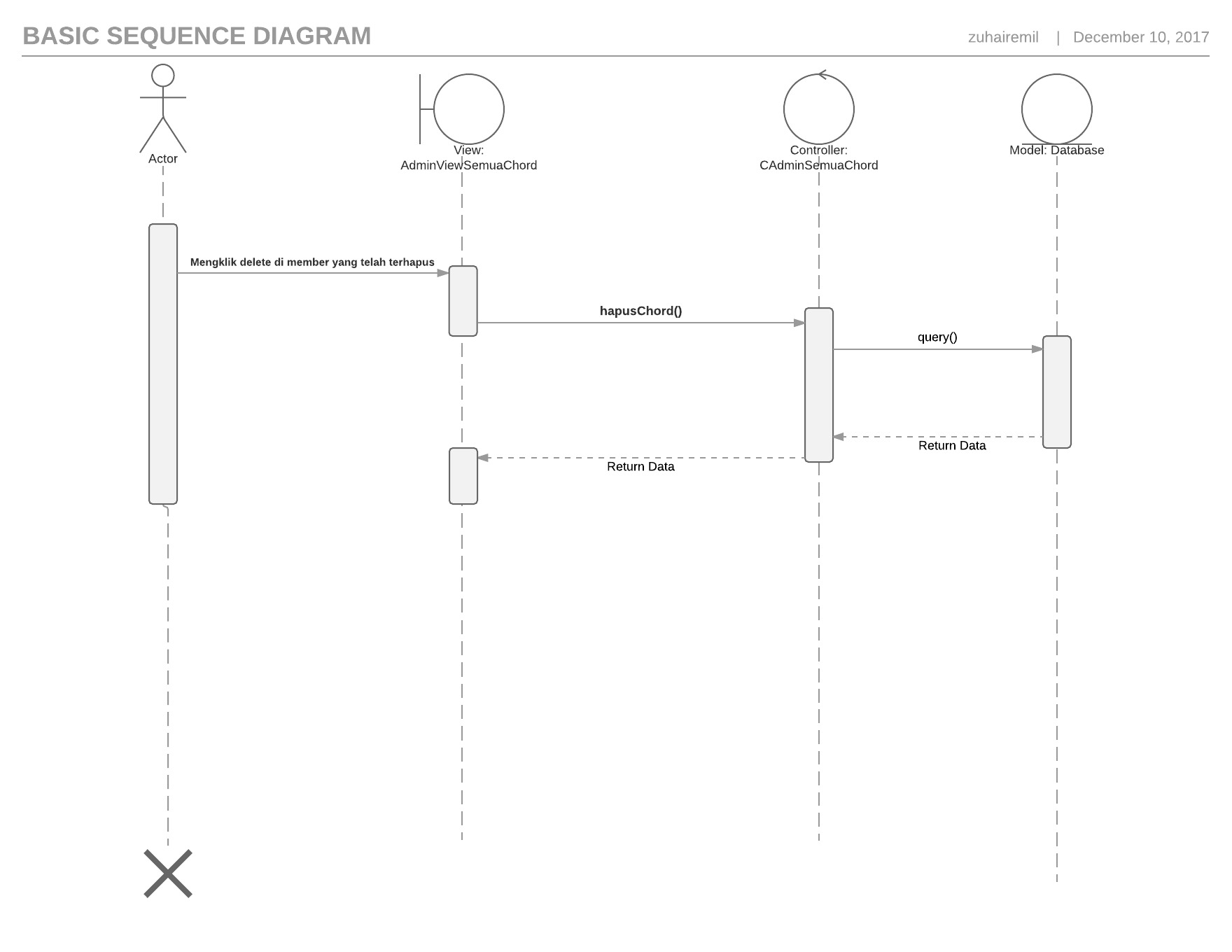


Diagram 3.4.3.25 Hapus Komentar Member

#### Pulihkan Akun Member (CHORON-F-026)

Untuk pulihkan akun member, admin harus mengklik pulihkan di salah satu member pada view AdminViewSemuaMember. Controller CAdminSemuaMember akan menjalankan fungsi pulihAkun() yaitu menjalankan query untuk pulihkan akun member.

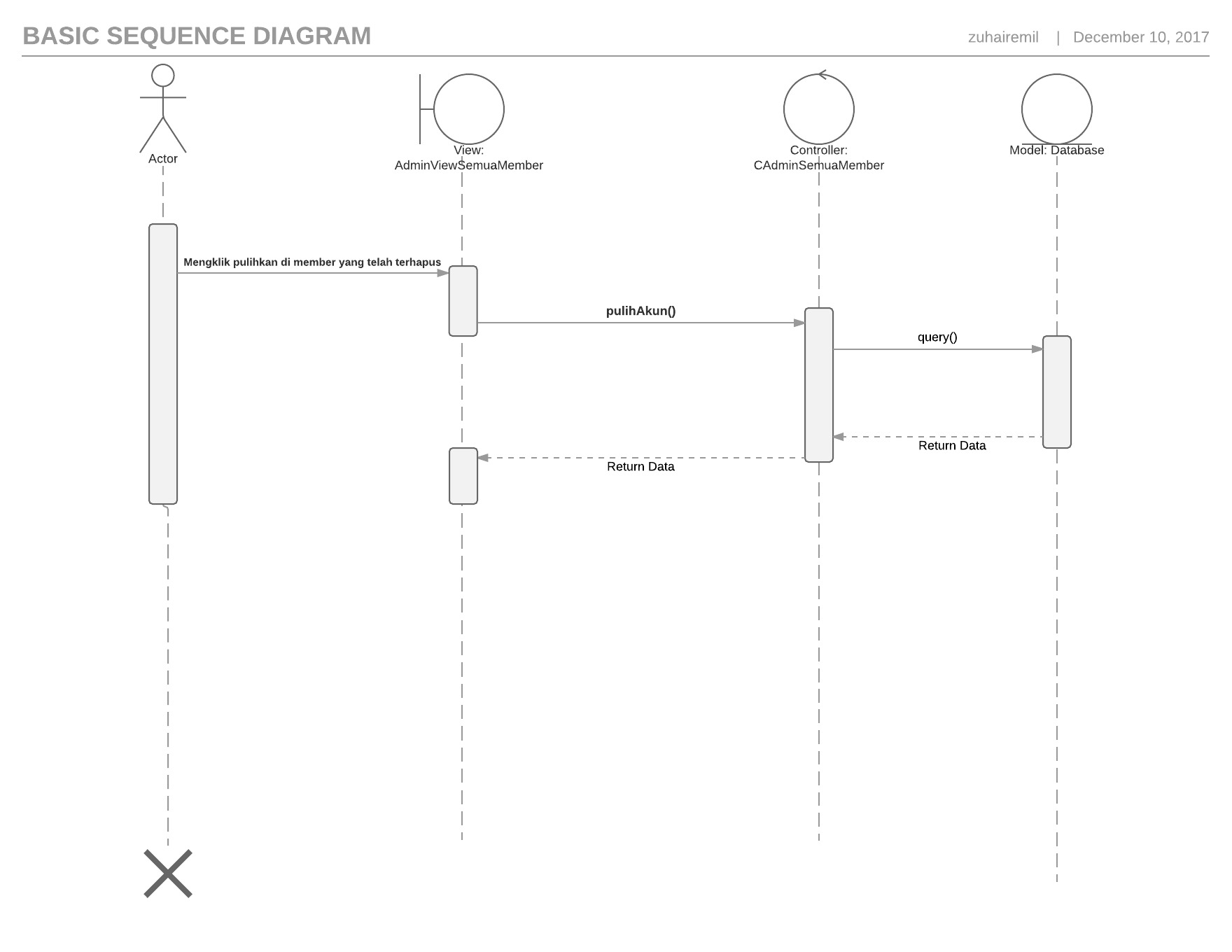


Diagram 3.4.3.26 Hapus Akun Member

#### Pulihkan Chord Member (CHORON-F-027)

Untuk pulihkan chord member, admin harus mengklik pulihkan di salah satu member pada view AdminViewSemuaChord. Controller CAdminSemuaMember akan menjalankan fungsi pulihChord() yaitu menjalankan query untuk pulihkan chord member.

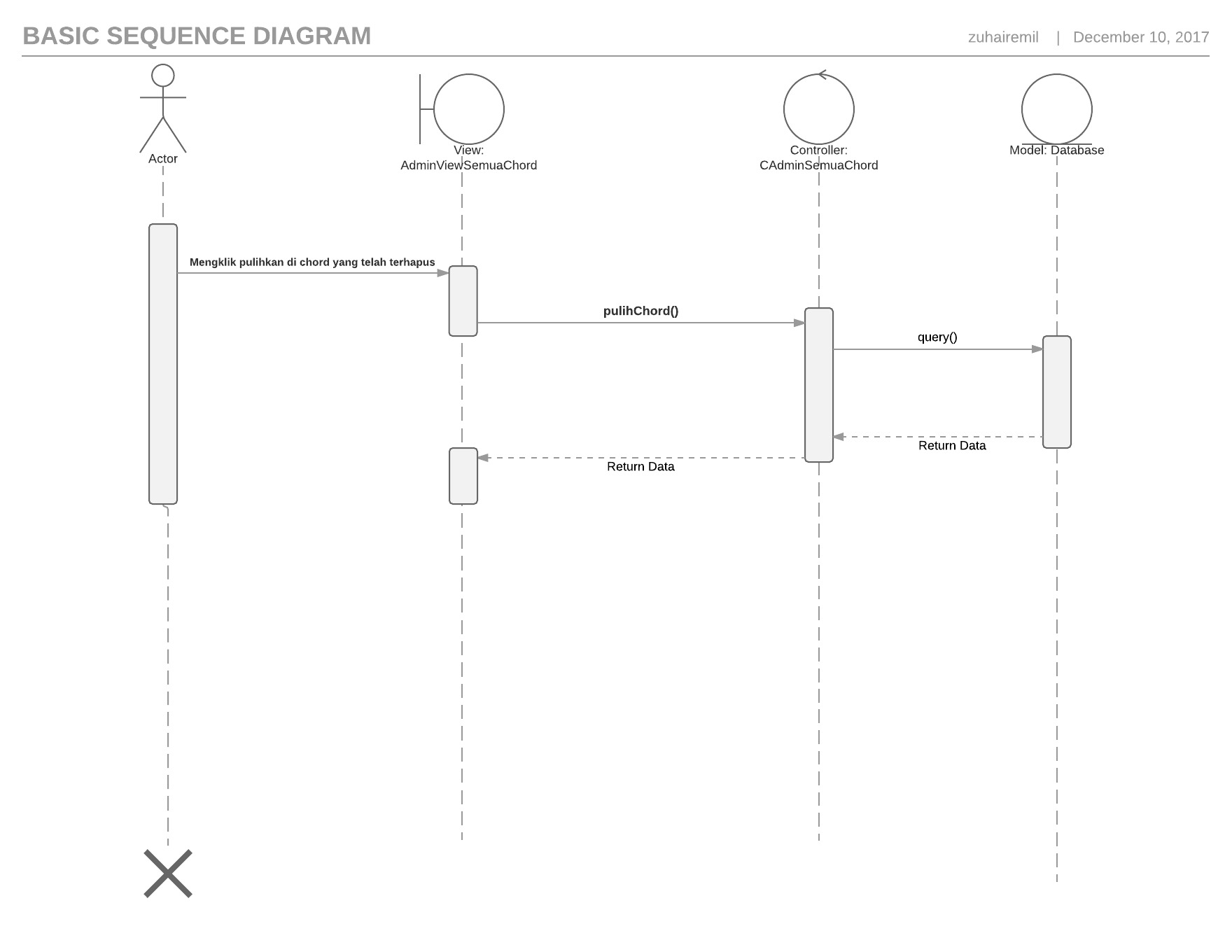


Diagram 3.4.3.27 Pulihkan Chord Member

#### Hapus Komentar Member (CHORON-F-028)

Untuk pulihkan komentar member, admin harus mengklik pulihkan di salah satu komentar pada view AdminViewSemuaKomentar. Controller CAdminSemuaKomentar akan menjalankan fungsi pulihKomentar() yaitu menjalankan query untuk pulihkan komentar member.

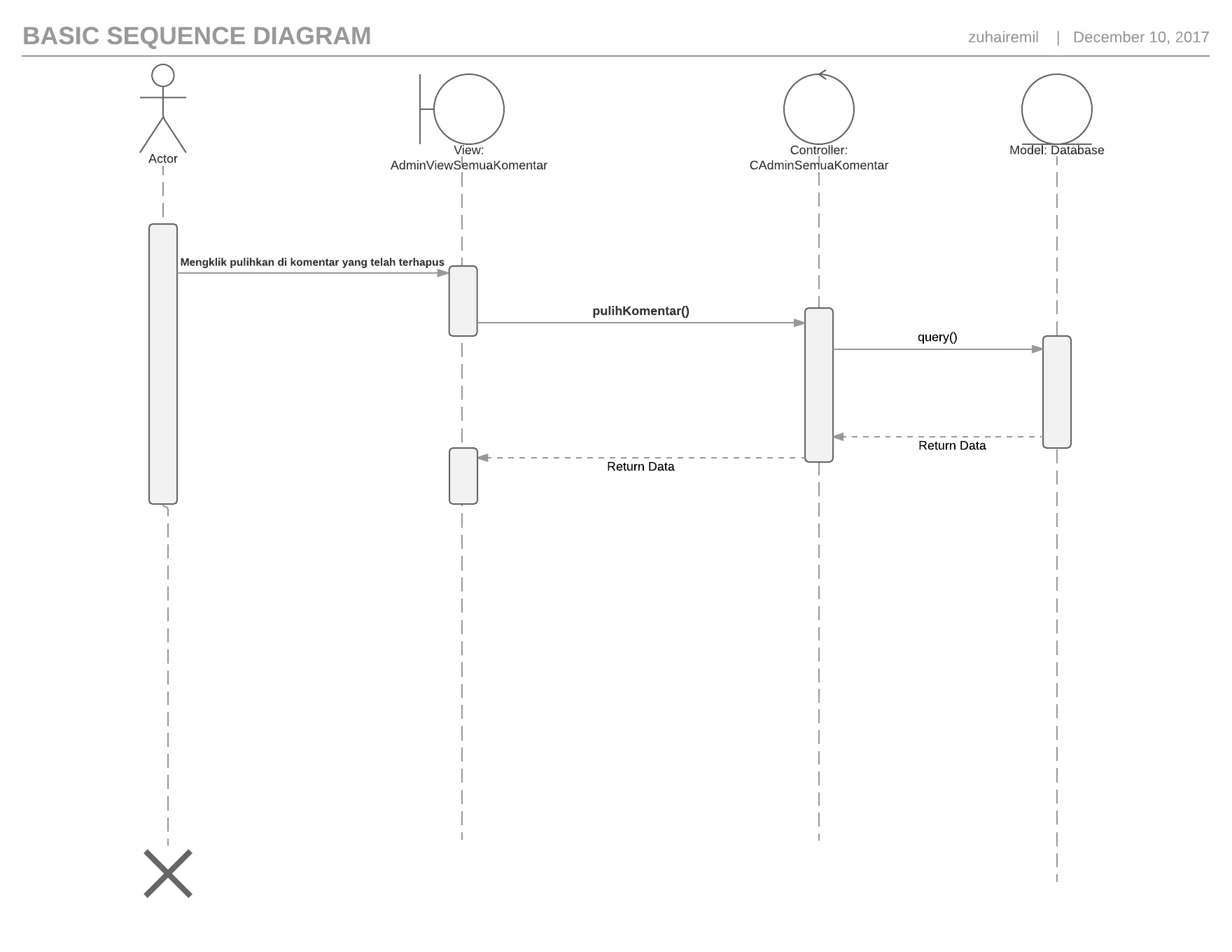


Diagram 3.4.3.28 Pulih Komentar Member

### Class Diagram

Class boundary yang bernama WebInterface memiliki hubungan agregasi dengan class Controller. Class control memiliki hubungan agregasi dengan class boundary dan class model serta hubungan generalisasi dengan class CI\_Controller. Class model memiliki hubungan agregasi dengan class control dan memiliki hubungan generalisasi dengan class CI\_Model.

### 

Diagram 3.4.4 Class Diagram