

SKRIPSI

IT STUDENT PORTAL: PEMANFAATAN *WEB SCRAPING*
UNTUK KUSTOMISASI PORTAL AKADEMIK MAHASISWA



HERFAN HERYANDI

NPM: 2012730012

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2015

UNDERGRADUATE THESIS

**IT STUDENT PORTAL: EXPLOITING WEB SCRAPING FOR
PORTAL AKADEMIK MAHASISWA CUSTOMIZATION**



HERFAN HERYANDI

NPM: 2012730012

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY**

2015

DAFTAR ISI

DAFTAR ISI	v
DAFTAR GAMBAR	vi
DAFTAR TABEL	viii
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan	3
1.4 Batasan Masalah	3
1.5 Metode Penelitian	3
1.6 Sistematika Penulisan	3
2 DASAR TEORI	5
2.1 jsoup	5
2.1.1 Jsoup	5
2.1.2 Connection	5
2.1.3 Response	6
2.1.4 Document	7
2.1.5 Elements	7
2.1.6 Element	7
2.2 Chrome DevTools	8
2.2.1 Elements	9
2.2.2 Network	9
2.3 Play Framework	14
2.4 SIA Models	16
2.5 CSS <i>Selector</i>	21
3 ANALISIS	23
3.1 Analisis Portal Akademik Mahasiswa	23
3.2 Analisis Kebutuhan IT Student Portal	33
3.3 Analisis Komunikasi Portal Akademik Mahasiswa untuk Fitur IT Student Portal	36
3.3.1 Kasus <i>Login</i>	37
3.3.2 Kasus Nilai	39
3.3.3 Kasus Jadwal	42
3.3.4 Kasus <i>Logout</i>	47
3.4 Analisis <i>Use Case</i>	49
3.4.1 Diagram <i>Use Case</i>	49
3.4.2 Skenario <i>Use Case</i>	50
3.5 Analisis Kelas	51
DAFTAR REFERENSI	55

DAFTAR GAMBAR

1.1	Prasyarat Mata Kuliah[1]	2
1.2	Prasyarat Mata Kuliah Prtal Akademik Mahasiswa[2]	2
2.1	Chrome DevTools	8
2.2	Panel Elements	9
2.3	Panel Network	10
2.4	Contoh Tautan Header pada Metode Permintaan GET	11
2.5	Contoh Tautan Header pada Metode Permintaan POST	11
2.6	Contoh <i>Preview</i> yang Tersedia	12
2.7	Contoh <i>Preview</i> yang Tidak Tersedia	12
2.8	Contoh Tautan Response	13
2.9	Contoh Tabel pada Tautan Cookie	14
2.10	Struktur Aplikasi Play Framework	14
2.11	Contoh Komponen Route[3]	15
2.12	Direktori app yang Dibangkitkan Play Framework[3]	15
2.13	Contoh <i>Method</i> pada Kelas <i>Controller</i>	15
2.14	Penerimaan <i>Parameter</i> pada <i>View</i>	16
2.15	Format Penulisan Definisi CSS	21
3.1	Halaman Utama Portal Akademik Mahasiswa	23
3.2	Menu Atas Home	24
3.3	Menu Atas Kuliah	24
3.4	Menu Atas Profil	24
3.5	Menu Atas Komentar	25
3.6	Identitas Portal	25
3.7	Menu Utama	25
3.8	Tampilan Registrasi FRS/PRS	26
3.9	Tampilan Kartu Rencana Studi[4]	26
3.10	Tampilan Pindah Kelas MKU[4]	27
3.11	Tampilan Jadwal Kuliah, UTS, dan UAS	27
3.12	Tampilan Jadwal MKU	28
3.13	Tampilan Jadwal Seluruh Fakultas	28
3.14	Tampilan Riwayat Per Semester	29
3.15	Tampilan Daftar Perkembangan Studi	29
3.16	Tampilan Statistik Nilai dan IP	30
3.17	Tampilan Riwayat Indeks Prestasi	30
3.18	Tampilan TOEFL	31
3.19	Tampilan Pembayaran Uang Kuliah	31
3.20	Tampilan Informasi	32
3.21	Tampilan <i>Pop Up</i> Informasi	32
3.22	Tampilan Kalender	32
3.23	Tampilan Info Browser	32
3.24	Tampilan Jadwal pada Portal Akademik Mahasiswa	34

3.25 Tampilan Jadwal yang Diinginkan Mahasiswa	34
3.26 Tampilan <i>Mobile</i> Portal Akademik Mahasiswa	35
3.27 Tombol “ <code>input#submit.login-button</code> ” pada Halaman Depan Portal Akademik Mahasiswa	37
3.28 Kolom “ <code>Username</code> ”“ <code>input#username.required</code> ” pada Halaman CAS UNPAR	38
3.29 <i>Form Data</i> yang dikirim CAS UNPAR	39
3.30 Elemen “a” dengan teks “Riwayat Per Semester” pada Menu Nilai dan Indeks Prestasi	40
3.31 <i>Combo Box</i> “ <code>select#tahun_akd_sec</code> ” pada Halaman Riwayat Per Semester	41
3.32 <i>Form Data</i> pada pengiriman Nilai Seluruh Tahun Akademik	42
3.33 Elemen “a” dengan teks “Kuliah, UTS dan UAS” pada Menu Jadwal	43
3.34 <i>Combo Box</i> “ <code>select#tahun_akd_sec</code> ” pada Halaman Jadwal Kuliah, UTS, dan UAS	44
3.35 <i>Form Data</i> pada pengiriman Jadwal Kuliah dan Ujian	45
3.36 Elemen “a” dengan teks “Seluruh Fakultas” pada Menu Jadwal	45
3.37 <i>Combo Box</i> “ <code>select#jadwal_all_ps</code> ” pada Halaman Jadwal Seluruh Fakultas	46
3.38 <i>Form Data</i> pada pengiriman Jadwal Seluruh Fakultas	47
3.39	47
3.40 Pengadaluarsaan Cookie CASTGC dan CASPRIVACY	48
3.41 Pengalihan ke Halaman Depan Portal Akademik Mahasiswa	49
3.42 Diagram <i>Use Case</i> IT Student Portal	49
3.43 Diagram Kelas Analisis IT Student Portal	52

DAFTAR TABEL

2.1 Tabel Rincian Kelas pada <i>Package id.ac.unpar.siamodels.matakuliah</i>	21
3.1 Tabel Hasil Analisis Kebutuhan IT Student Portal	36

¹

BAB 1

²

PENDAHULUAN

³ 1.1 Latar Belakang

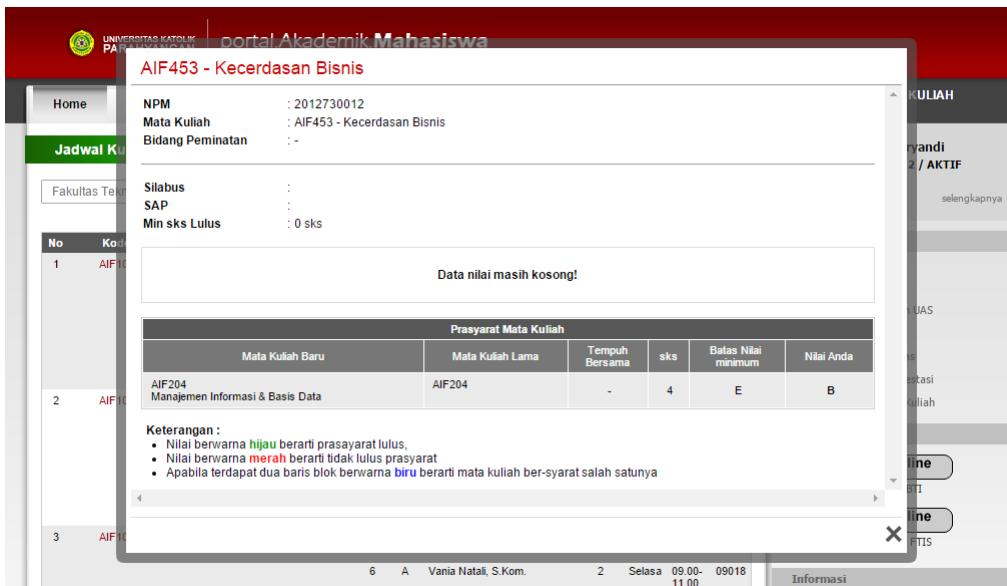
⁴ Portal Akademik Mahasiswa atau dikenal sebagai Student Portal UNPAR^[2] merupakan
⁵ sistem informasi berbasis web yang digunakan oleh mahasiswa Universitas Katolik Para-
⁶ hyangan. Fitur-fitur yang dimiliki Prtal Akademik Mahasiswa yaitu rencana studi, jadwal,
⁷ nilai dan indeks prestasi, dan pembayaran uang kuliah. Namun, fitur-fitur tersebut ma-
⁸ sih belum cukup untuk mendukung kebutuhan akademik mahasiswa Program Studi Teknik
⁹ Informatika.

¹⁰ Salah satu fitur yang diperlukan oleh mahasiswa Teknik Informatika UNPAR adalah
¹¹ prasyarat mata kuliah. Dalam Teknik Informatika UNPAR, terdapat beberapa mata kuliah
¹² yang membutuhkan prasyarat baik prasyarat tempuh maupun prasyarat lulus. Prtal Akade-
¹³ mik Mahasiswa sudah menyediakan fitur prasyarat mata kuliah namun kurang mendukung
¹⁴ karena data yang ditampilkan kurang akurat. Misalnya dalam pengambilan mata kuliah
¹⁵ “AIF453 Kecerdasan Bisnis” membutuhkan prasyarat lulus mata kuliah “AIF204 Manaje-
¹⁶ men Informasi dan Basis Data” atau lulus mata kuliah “AIF102 Algoritma dan Struktur
¹⁷ Data” dengan IPK di atas 2.75 (Gambar 1.1). Namun dalam Portal Akademik Mahasiswa,
¹⁸ prasyarat yang dicantumkan hanya lulus mata kuliah “AIF204 Manajemen Informasi dan
¹⁹ Basis Data” (Gambar 1.2). Selain itu, pemeriksaan prasyarat mata kuliah tidak dilakukan
²⁰ secara otomatis sehingga setiap pengambilan mata kuliah tetap dianggap valid meskipun
²¹ belum memenuhi prasyarat.

²² jsoup^[5] merupakan *library* Java yang digunakan untuk menelusuri suatu situs web untuk
²³ mendapatkan suatu informasi. Informasi yang didapat berupa HTML yang kemudian dieks-
²⁴ trak dan disajikan dalam bentuk *Document Object Model*. Play Framework^[3] merupakan
²⁵ sebuah *web framework* berbasis Java dan Scala. Play juga menggunakan *design pattern*
²⁶ *Model-View-Controller* (MVC) di mana *model* dan *controller* menggunakan bahasa Java se-
²⁷ dangkan *view* menggunakan bahasa Scala dan HTML. SIA Models^[6] merupakan kelas-kelas
²⁸ dalam bahasa Java yang merepresentasikan Sistem Informasi Akademik UNPAR. Aplikasi
²⁹ akan dibuat dengan menggunakan Play Framework dan jsoup karena aplikasi didukung oleh
³⁰ SIA Models yang tersedia dalam bahasa Java.

Prasyarat Mata Kuliah					
4	AIF203	Struktur Diskret	1. T : AIF103 Matematika Diskret		
5	AIF205	Arsitektur & Organisasi Komputer	1. T : AIF106 Sistem Digital		
6	AIF301	Pengantar Sistem Cerdas	1. T : AIF104 Logika Informatika 2. T : AIF202/292 Desain dan Analisis Algoritma		
4	AIF302	Penulisan Ilmiah	1. Lulus minimal 84 SKS		
5	AIF303	Pengantar Sistem Informasi	1. T : AIF204/AIF294 Manaj. Informasi & Basis Data		
6	AIF305	Jaringan Komputer	1. T : AIF206 Sistem Operasi		
	AIF403	Komputer dan Masyarakat	1. Sudah lulus minimal 72 SKS		
7	AIF405	Proyek Sistem Informasi 2	1. T : AIF304 Proyek Sistem Informasi 1		
8	AIF401	Skripsi 1	1. L : AIF302 Penulisan Ilmiah 2. Lulus minimal 108 SKS		
9	AIF402	Skripsi 2	1. L : AIF401 Skripsi 1 ATAU 2. Ambil AIF401 Skripsi 1 & lulus min. 124 SKS		
13	APS402	Etika Profesi	1. Sudah lulus minimal 90 SKS		
Kuliah Pilihan Wajib					
1	AIF311	Pemrograman Fungsional	1. T : AIF103 Matematika Diskret		
2	AIF314	Pemrograman Basisdata	1. T : AIF204/AIF294 Manajemen Informasi & Basis Data		
3	AIF315	Pemrograman Berbasis Web	2. AIF204/AIF294 Manaj. Informasi & Basis Data		
4	AIF317	Desain Antarmuka Grafis	1. T : AIF210 Interaksi Manusia Komputer		
Kuliah Pilihan					
1	AIF438	Penambangan Data	1. T : AIF102/AIF192 Algoritma & Struktur Data 1. T : AIF204/AIF 294 Manaj. Inf. & Basis Data		
2	AIF453	Kecerdasan Bisnis	ATAU 2. T : AIF102/AIF 192 Algo. & Struk. Data & IPK >2.75		
3	AIF457	Kewirausahaan Berbasis Teknologi	1. Lulus minimal 70 SKS		
4	AIF469	Layanan Berbasis Web	1. T : AIF305 Jaringan Komputer 2. T : AIF315 Pemrograman Berbasis Web		
5	AIF362	Sistem & Aplikasi Telematika	1. T : AIF305 Jaringan Komputer		

Gambar 1.1: Prasyarat Mata Kuliah[1]



Gambar 1.2: Prasyarat Mata Kuliah Prtal Akademik Mahasiswa[2]

- Untuk mendukung kebutuhan akademik mahasiswa Program Studi Teknik Informatika, fitur-fitur yang diperlukan akan dianalisa kemudian diimplementasikan ke dalam program IT Student Portal. Program yang akan dibuat merupakan program berbasis web menggunakan Play Framework. Selain itu, data-data yang akan ditampilkan diambil langsung dari Prtal Akademik Mahasiswa dengan *web scraping* menggunakan *library* jsoup. Untuk melakukan pengambilan data, jsoup harus mengetahui cara kerja dari Prtal Akademik Mahasiswa. Analisis komunikasi Prtal Akademik Mahasiswa akan dilakukan dengan menggunakan Chrome DevTools.

1 1.2 Rumusan Masalah

2 Rumusan dari masalah yang akan dibahas pada skripsi ini adalah sebagai berikut:

- 3 1. Fitur-fitur apa saja yang akan dibuat untuk IT Student Portal?
- 4 2. Bagaimana mengimplementasikan *web scraping* menggunakan *library jsoup*?
- 5 3. Bagaimana membangun aplikasi IT Student Portal?

6 1.3 Tujuan

7 Tujuan-tujuan yang hendak dicapai pada skripsi ini adalah sebagai berikut:

- 8 1. Mengetahui fitur-fitur yang akan dibuat dalam IT Student Portal.
- 9 2. Mengimplementasikan *web scraping* menggunakan *library jsoup*.
- 10 3. Membangun aplikasi IT Student Portal.

11 1.4 Batasan Masalah

12 Beberapa batasan yang dibuat terkait dengan pengerjaan skripsi ini adalah sebagai berikut:

- 13 1. Prasyarat mata kuliah yang tersedia hanya mata kuliah yang didukung SIA Models.
- 14 2. Aplikasi akan diuji pada server FTIS sehingga tidak bisa diakses dari luar jaringan FTIS.

16 1.5 Metode Penelitian

17 Metode-metode yang dilakukan pada skripsi ini adalah sebagai berikut:

- 18 1. Melakukan studi mengenai *library jsoup*, Chrome DevTools, dan Play Framework.
- 19 2. Melakukan wawancara.
- 20 3. Menganalisis Portal Akademik Mahasiswa.
- 21 4. Mengimplementasikan *web scraping* menggunakan *library jsoup*.
- 22 5. Melakukan eksperimen dan pengujian.

23 1.6 Sistematika Penulisan

24 Sistematika penulisan setiap bab pada skripsi ini adalah sebagai berikut:

- 25 1. Bab Pendahuluan
26 Bab 1 berisi latar belakang, rumusan masalah, tujuan, metode penelitian, dan sistematika penulisan yang digunakan untuk menyusun skripsi ini.
- 27

1 2. Bab Dasar Teori

2 Bab 2 berisikan teori-teori yang digunakan dalam pembuatan skripsi ini. Teori yang
3 digunakan yaitu *library jsoup*, Chrome DevTools, Play Framework, SIA Models, dan
4 CSS *selector*.

5 3. Bab Analisis

6 Bab 3 berisikan analisis yang dilakukan pada skripsi ini, meliputi analisis Portal Aka-
7 demik Mahasiswa, analisis kebutuhan IT Student Portal, analisis komunikasi Portal
8 Akademik Mahasiswa untuk fitur IT Student Portal, analisis *use case* meliputi diagram
9 *use case* dan skenario, serta analisis kelas.

¹

BAB 2

²

DASAR TEORI

³ 2.1 jsoup

⁴ *Web scraping* adalah teknik mendapatkan informasi dari sebuah situs web secara otomatis^[7].
⁵ Dalam bahasa Java, *web scraping* dapat diimplementasikan menggunakan *library* jsoup^[5].
⁶ API yang disediakan oleh jsoup dapat digunakan untuk mengekstrak dan memanipulasi data
⁷ HTML.

⁸ Subbab-subbab berikut menjelaskan beberapa kelas dari jsoup.

⁹ 2.1.1 Jsoup

¹⁰ Kelas ini merupakan inti untuk mengakses fungsi jsoup. Seluruh *method* dalam kelas ini
¹¹ merupakan **static method** sehingga kelas ini tidak perlu dikonstruksi. Salah satu *method*
¹² yang dimiliki kelas ini adalah sebagai berikut:

- ¹³ • **public static Connection connect(String url)**

¹⁴ Berfungsi untuk membuat koneksi baru dengan suatu situs web.

¹⁵ **Parameter:**

- ¹⁶ – **url** URL situs web dengan protokol HTTP atau HTTPS.

¹⁷ **Kembalian:** koneksi dengan situs web.

¹⁸ 2.1.2 Connection

¹⁹ Kelas ini merupakan **interface** yang menyediakan pengambilan data dari situs web. Bebe-
²⁰ rapa *method* yang dimiliki kelas ini adalah sebagai berikut:

- ²¹ • **Connection cookies(Map<String, String> cookies)**

²² Berfungsi untuk menambahkan *cookie*.

²³ **Parameter:**

- ²⁴ – **cookies** Map dari *cookie*.

²⁵ **Kembalian:** koneksi yang sama tetapi sudah diubah.

- ²⁶ • **Connection data(String key, String value)**

²⁷ Berfungsi untuk menambahkan parameter data yang bisa dikirim melalui metode HT-
²⁸ TP GET atau POST.

²⁹ **Parameter:**

- 1 – **key** kunci data.
- 2 – **value** nilai data.

3 **Kembalian:** koneksi yang sama tetapi sudah diubah.

- 4 • **Connection method(Connection.Method method)**

5 Berfungsi untuk mengatur metode permintaan HTTP, GET atau POST. Metode pe-
6 ngiriman secara *default* adalah GET

7 **Parameter:**

- 8 – **method** metode pengiriman permintaan HTTP.

9 **Kembalian:** koneksi yang sama tetapi sudah diubah.

- 10 • **Connection timeout(int millis)**

11 Berfungsi untuk mengatur batas waktu *request*. Batas waktu nol akan dianggap seba-
12 gai batas waktu yang tak terhingga.

13 **Parameter:**

- 14 – **millis** batas waktu dalam milidetik.

15 **Kembalian:** koneksi yang sama tetapi sudah diubah.

- 16 • **Connection validateTLCertificates(boolean value)**

17 Berfungsi untuk mengatur pemeriksaan sertifikat TLS untuk permintaan HTTPS. Ni-
18 lai **true** untuk memeriksa dan nilai **false** untuk tidak memeriksa.

19 **Parameter:**

- 20 – **value** status pemeriksaan sertifikat TLS.

21 **Kembalian:** koneksi yang sama tetapi sudah diubah.

- 22 • **Connection.Response execute()**

23 Berfungsi untuk mengirim permintaan HTTP.

24 **Kembalian:** objek Response.

25 **2.1.3 Response**

26 Kelas ini merepresentasikan permintaan HTTP. Beberapa *method* yang dimiliki kelas ini
27 adalah sebagai berikut:

- 28 • **Map<String, String> cookies()**

29 *Method* ini berfungsi untuk mendapatkan seluruh *cookies*.

30 **Kembalian:** seluruh *cookies*.

- 31 • **Document parse()**

32 Berfungsi untuk mengurai *body* jawaban menjadi dokumen.

33 **Kembalian:** koneksi yang sama tetapi sudah diubah.

- 34 • **String body()**

35 Berfungsi untuk mendapatkan *body* jawaban dalam bentuk *string*.

36 **Kembalian:** *body* jawaban dalam bentuk *string*.

1 **2.1.4 Document**

2 Kelas ini merepresentasikan dokumen HTML. Salah satu *method* yang dimiliki kelas ini
3 adalah sebagai berikut:

- 4 • **public Elements select(String cssQuery)**

5 *Method* ini diturunkan dari kelas Element, berfungsi untuk menemukan elemen HTML
6 yang sesuai dengan kueri CSS.

7 **Parameter:**

- 8 – **cssQuery** kueri CSS.

9 **Kembalian:** elemen-elemen HTML yang sesuai dengan kueri CSS.

10 **2.1.5 Elements**

11 Kelas ini merepresentasikan kumpulan elemen HTML. Beberapa *method* yang dimiliki kelas
12 ini adalah sebagai berikut:

- 13 • **public Elements select(String query)**

14 Berfungsi untuk menemukan elemen-elemen yang sesuai dalam *list* elemen.

15 **Parameter:**

- 16 – **query** kueri CSS.

17 **Kembalian:** elemen-elemen yang sudah diseleksi sesuai kueri.

- 18 • **public String val()**

19 Berfungsi untuk mendapatkan nilai dari elemen pertama.

20 **Kembalian:** nilai elemen.

- 21 • **public String text()**

22 *Method* Berfungsi untuk mendapatkan kombinasi teks dari seluruh elemen yang sesuai.

23 **Kembalian:** seluruh teks dalam *string*.

24 **2.1.6 Element**

25 Kelas ini merepresentasikan sebuah elemen HTML yang berisikan *tag*, atribut, dan anak
26 elemen. Beberapa *method* yang dimiliki kelas ini adalah sebagai berikut:

- 27 • **public Element child(int index)**

28 Berfungsi untuk mendapatkan anak elemen berdasarkan nomor indeks.

29 **Parameter:**

- 30 – **index** nomor index.

31 **Kembalian:** anak elemen.

- 32 • **public Element children()**

33 Berfungsi untuk mendapatkan seluruh anak elemen.

34 **Kembalian:** seluruh anak elemen.

- 1 • **public String className()**

2 Berfungsi untuk mendapatkan nama kelas elemen.

3 **Kembalian:** nama kelas elemen.

- 4 • **public String text()**

5 Berfungsi untuk mendapatkan teks dari elemen.

6 **Kembalian:** teks dalam *string*.

7 2.2 Chrome DevTools

8 Chrome Developer Tools (DevTools) adalah perangkat *debugging* yang dimiliki Google Chrome[8].

9 Saat menunjungi suatu halaman web, pengguna DevTools dapat melakukan *debugging* pada 10 halaman tersebut. DevTools dapat diakses dengan menekan “Ctrl+Shift+I” saat sedang 11 membuka suatu halaman web.

12 Panel-panel yang dimiliki DevTools (Gambar 2.1) antara lain:

13 1. **Elements**, memeriksa dan mengubah elemen HTML dan *style* dari suatu situs web.

14 2. **Console**, mendapatkan informasi pengembangan dan berinteraksi dengan dokumen.

15 3. **Sources**, melakukan *debugging* pada JavaScript dengan menentukan *breakpoint*.

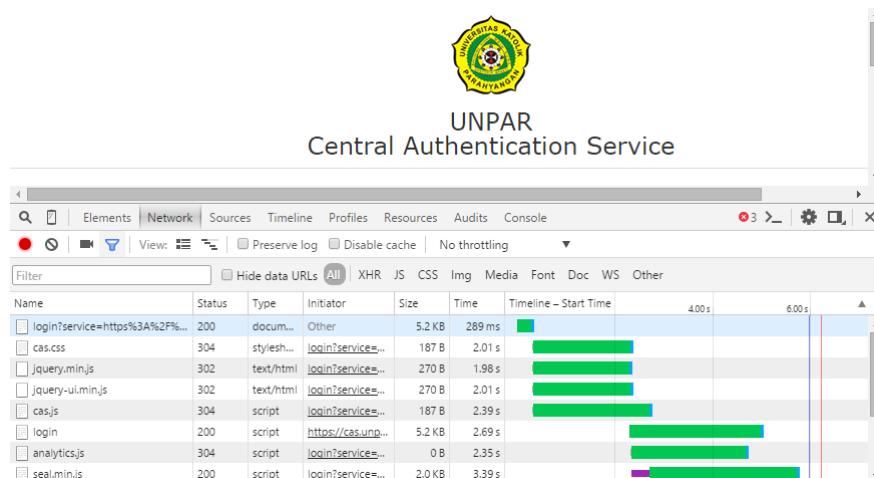
16 4. **Network**, memantau aktivitas jaringan pada situs web secara *real-time*.

17 5. **Audits**, menganalisa halaman yang dimuat.

18 6. **Timeline**, menampilkan alur waktu saat memuat halaman.

19 7. **Profiles**, menggambarkan waktu eksekusi dan penggunaan memori saat memuat halaman.

21 8. **Resources**, memeriksa sumber daya halaman yang dapat berupa basis data, *cookies*, dan *cache*.



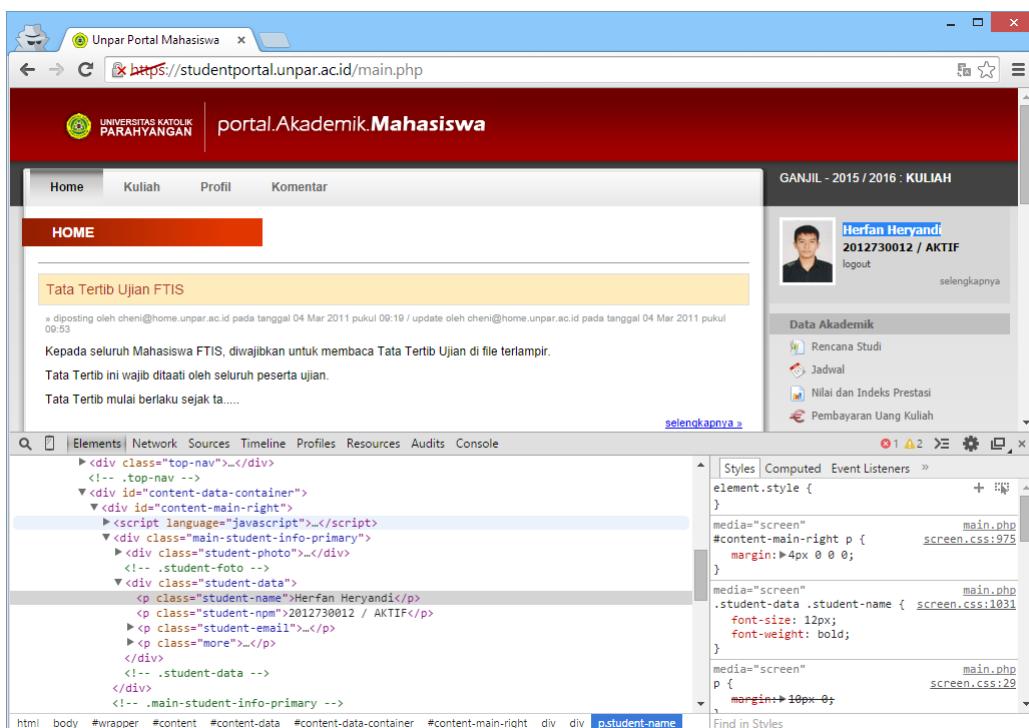
Gambar 2.1: Chrome DevTools

23 Pada subbab-subbab berikut akan dijelaskan mengenai dua panel dari DevTools.

2.2.1 Elements

Panel Elements memungkinkan untuk memperlihatkan informasi yang terstruktur tentang halaman yang sedang dibuka. HTML akan ditampilkan dalam bentuk pohon *Document Object Model* (DOM). DOM adalah sebuah struktur seperti pohon yang dibuat oleh browser untuk menemukan elemen HTML¹. Tampilan pohon DOM memperlihatkan struktur DOM dari halaman yang sedang dibuka. Pohon DOM adalah pohon dari node-node yang mewakili setiap elemen HTML seperti <body> dan <p>.

Pemeriksaan elemen akan memperlihatkan node DOM dan CSS dari elemen yang dipilih pada *browser*. Pemeriksaan elemen dapat dilakukan dengan cara klik kanan pada elemen yang ingin diperiksa kemudian pilih “Inspect element”. Dengan melakukan pemeriksaan elemen, jendela panel Elements akan muncul. Sebagai contoh pada gambar 2.2, saat melakukan “Inspect element” pada nama mahasiswa, panel Elements akan muncul dan menunjukkan pohon DOM dari halaman tersebut. Selain itu panel Elements juga menunjukkan CSS selector dari elemen tersebut yaitu p.student-name.



Gambar 2.2: Panel Elements

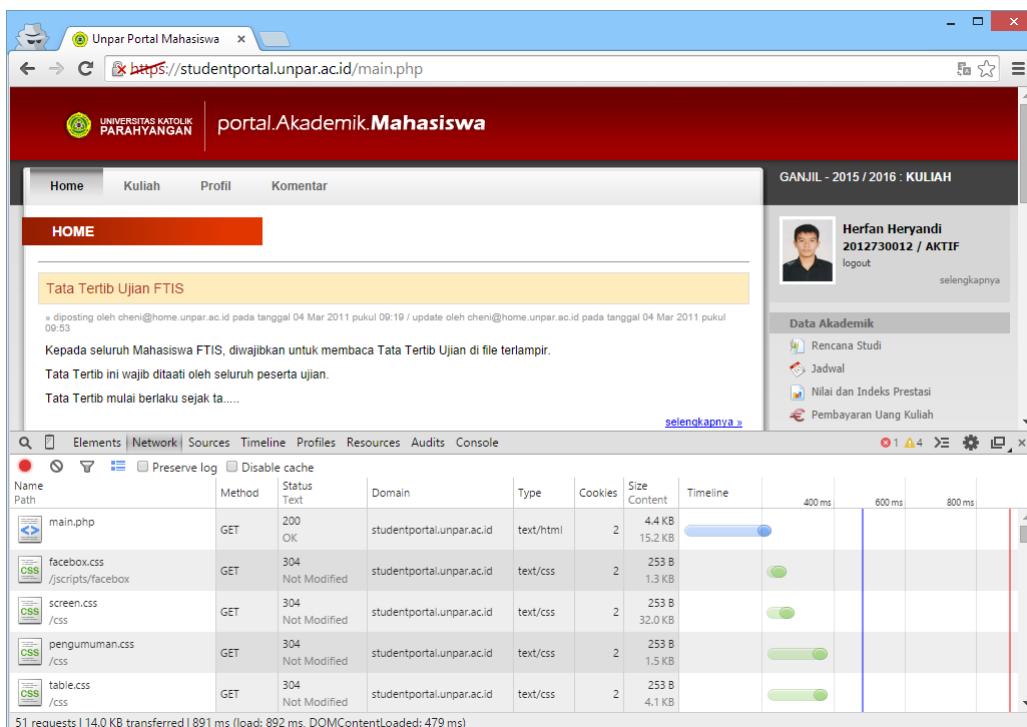
2.2.2 Network

Panel Network secara otomatis merekam semua aktivitas jaringan saat DevTools terbuka. Pertama kali dibuka, panel Network masih kosong. Halaman web harus dimuat ulang untuk mulai merekam aktivitas jaringan atau menunggu adanya aktivitas jaringan pada halaman web. Panel Network akan mencatat sumber daya dari aktivitas jaringan yang terekam. Setiap sumber daya akan ditambahkan ke dalam sebuah baris dalam tabel Network seperti pada gambar 2.3 dengan rincian kolom sebagai berikut:

- **Name dan Path**, nama dan URL dari sumber daya.

¹<http://try.jquery.com/>, diakses 24 September 2015

- 1 ● **Method**, metode permintaan HTTP.
- 2 ● **Status dan Text**, kode status HTTP dan pesan.
- 3 ● **Domain**, domain dari sumber daya.
- 4 ● **Type**, tipe sumber daya yang diminta.
- 5 ● **Cookies**, banyaknya *cookie* yang dikirim dalam permintaan.
- 6 ● **Size dan Content**, *size* merupakan ukuran dari *header* dan *body* jawaban yang dikirim server sedangkan *content* merupakan ukuran konten sumber daya.
- 7
- 8 ● **Timeline**, alur waktu dari seluruh aktivitas jaringan yang diminta.

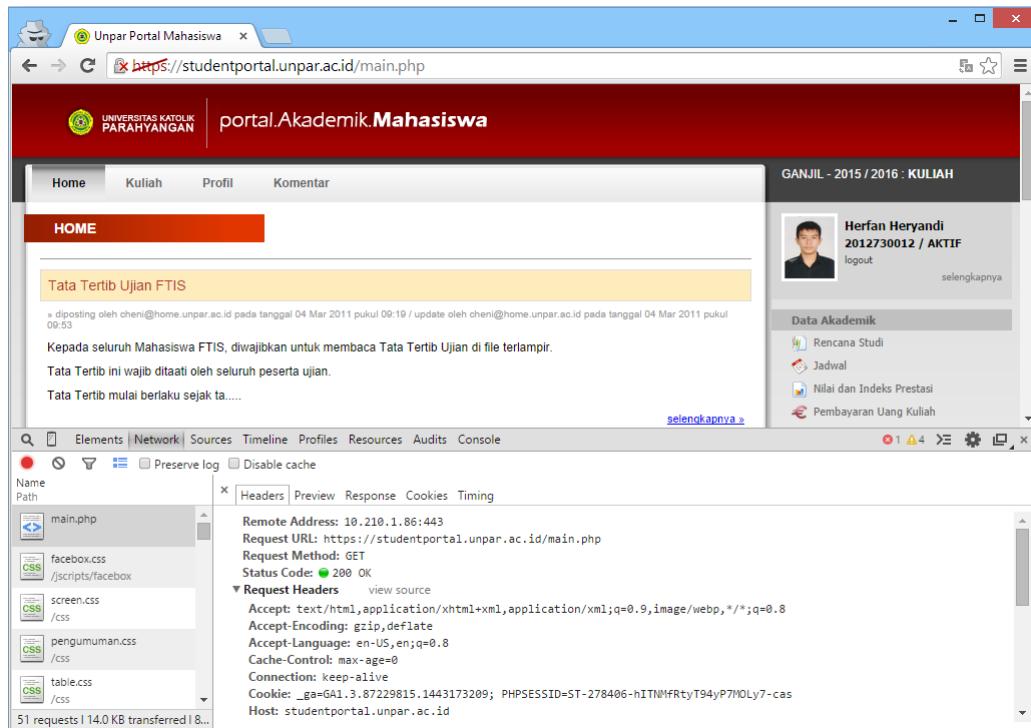


Gambar 2.3: Panel Network

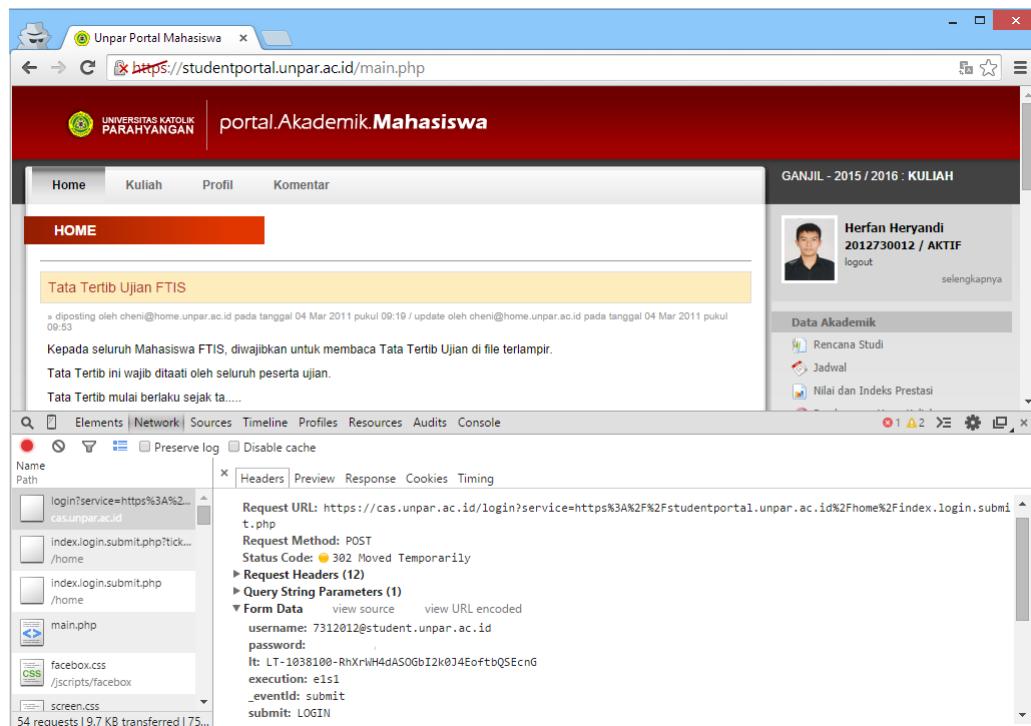
- 9 Ketika nama sumber daya dalam tabel Network diklik, maka akan muncul tautan baru
- 10 yang berisi rincian tambahan sebagai berikut:

11 ● **Header**

12 Tautan Header menampilkan *request URL*, *request method*, *status code*, HTTP *response* dan *request header* beserta nilainya, dan *query string parameter*. HTTP header dapat ditampilkan secara terformat atau dalam bentuk sumber dengan mengklik tombol *toggle* “view parsed”/“view source”. Nilai-nilai parameter dapat ditampilkan dalam bentuk yang sudah didekodekan atau dalam bentuk URL yang dienkode dengan mengklik tombol *toggle* “view decoded”/“view URL encoded”. Sebagai contoh pada gambar 2.4 menampilkan *header* pada metode permintaan GET sedangkan gambar 2.5 menampilkan *header* pada metode permintaan POST.



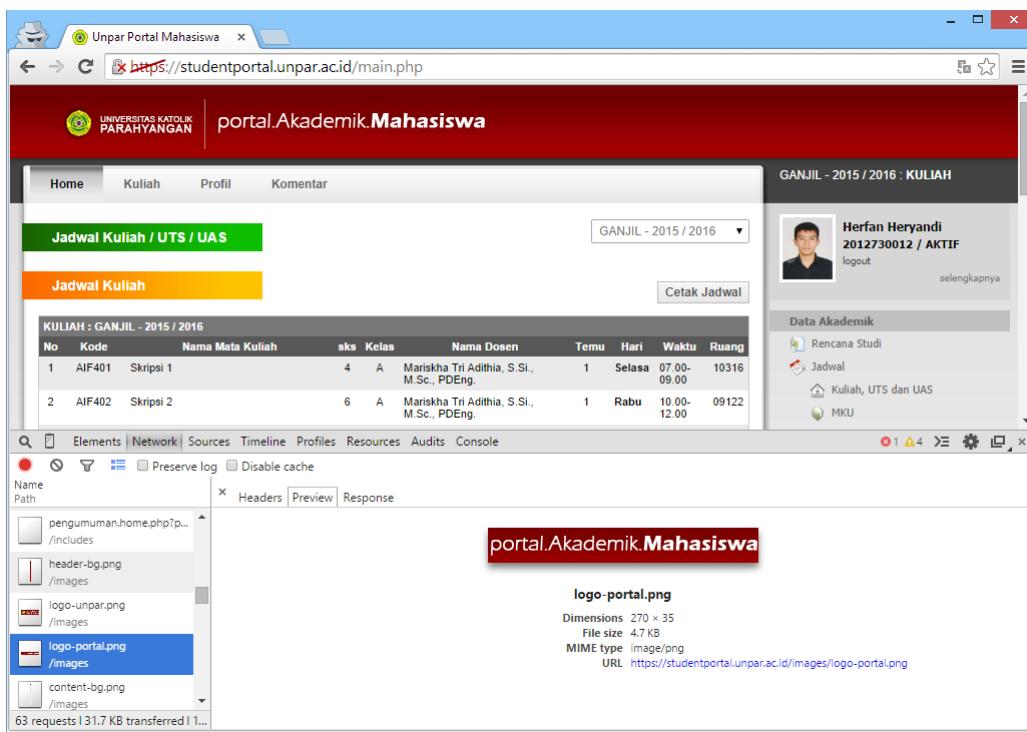
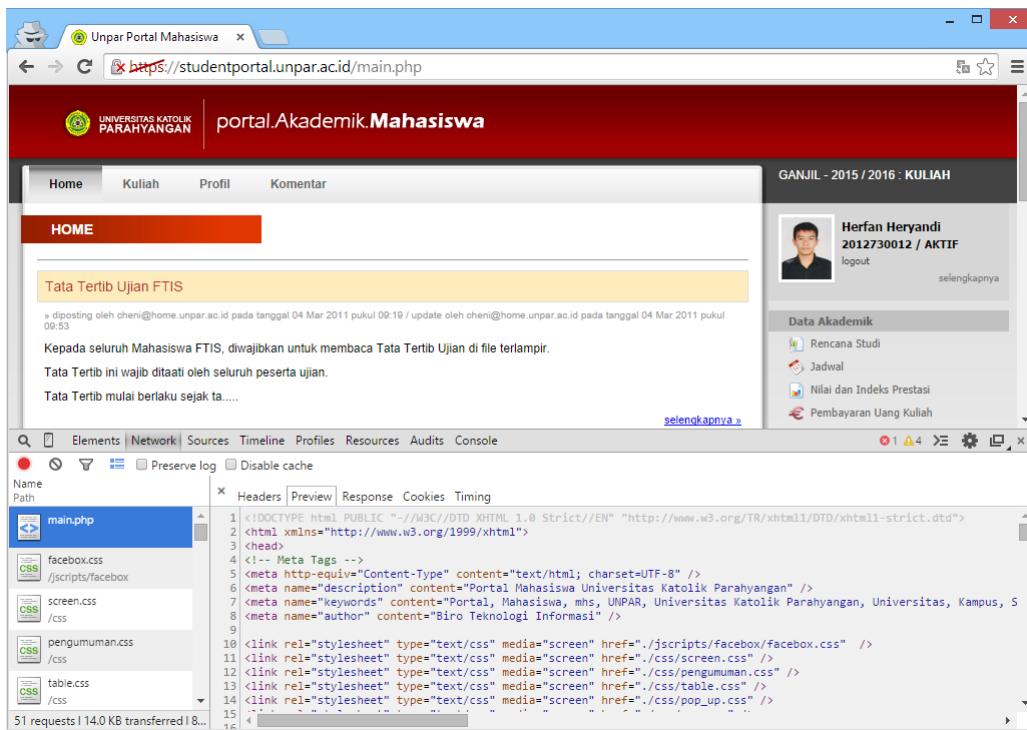
Gambar 2.4: Contoh Tautan Header pada Metode Permintaan GET



Gambar 2.5: Contoh Tautan Header pada Metode Permintaan POST

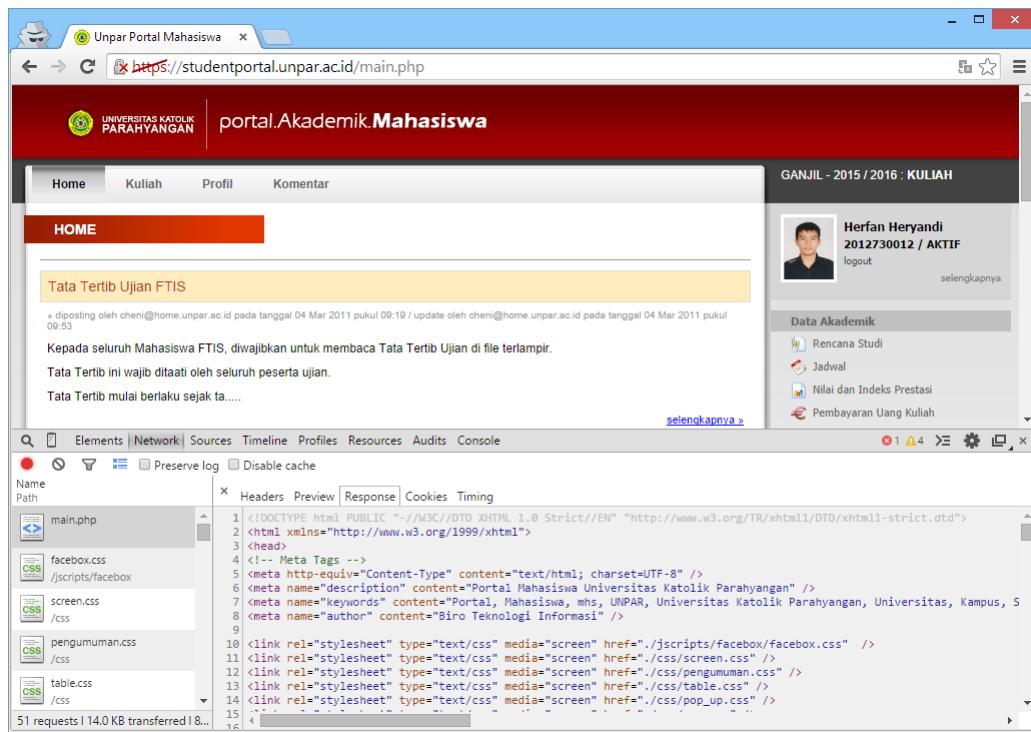
1. • Preview

Tautan Preview menampilkan *preview* sumber daya jika tersedia. Gambar 2.6 menampilkan *preview* yang tersedia pada sumber daya. Jika *preview* tidak tersedia maka akan tampilan akan sama dengan jawaban seperti yang terlihat pada gambar 2.7.

Gambar 2.6: Contoh *Preview* yang TersediaGambar 2.7: Contoh *Preview* yang Tidak Tersedia

• Response

- Tautan Response berisi konten sumber daya yang tidak terformat. Sebagai contoh pada gambar 2.8 menampilkan Tautan Response dari sumber daya `main.php`.

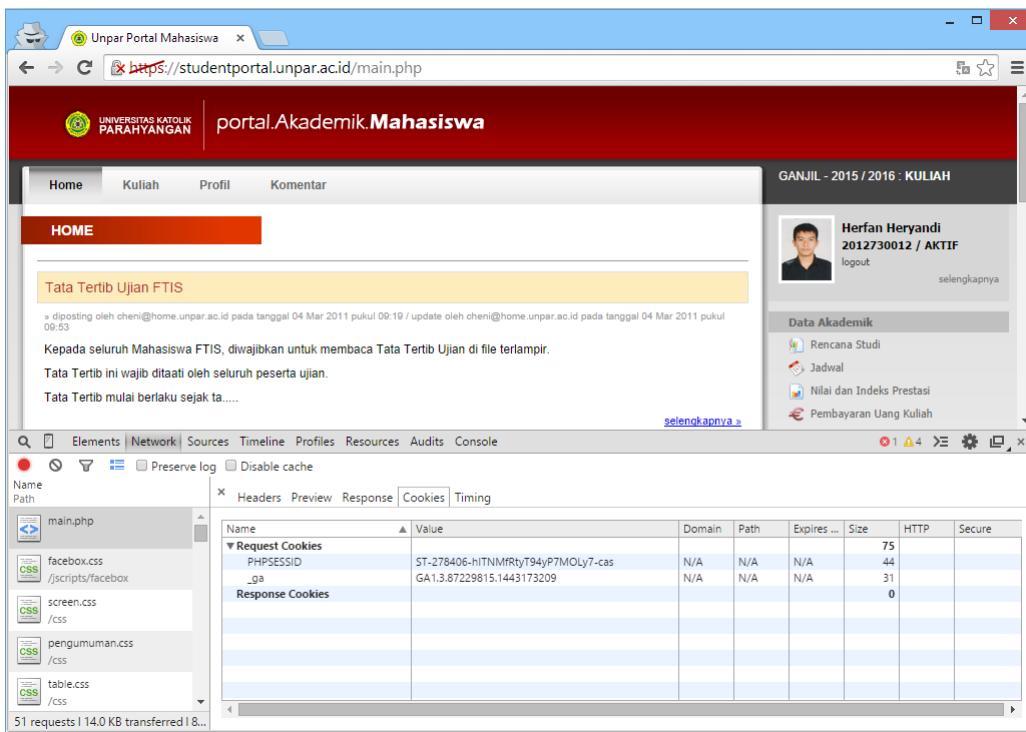


Gambar 2.8: Contoh Tautan Response

• Cookies

Tautan Cookies menampilkan sebuah tabel yang terdiri dari seluruh *cookie* yang di-transmisikan dalam *header* permintaan dan jawaban HTTP. Contoh dari tabel *cookie* dapat dilihat pada gambar 2.9 dengan rincian kolom sebagai berikut:

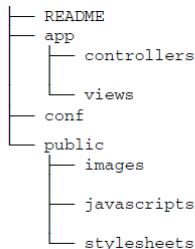
- **Name**, nama *cookie*
- **Value**, nilai *cookie*
- **Domain**, domain yang memiliki *cookie*
- **Path**, URL asal *cookie*
- **Expires/Max-Age**, batas akhir nilai *cookie*
- **Size**, ukuran *cookie* dalam byte
- **HTTP**, menunjukkan bahwa *cookie* harus ditetapkan oleh browser dalam permintaan HTTP, dan tidak dapat diakses dengan JavaScript
- **Secure**, menunjukkan bahwa *cookie* harus dikirim melalui koneksi yang aman



Gambar 2.9: Contoh Tabel pada Tautan Cookie

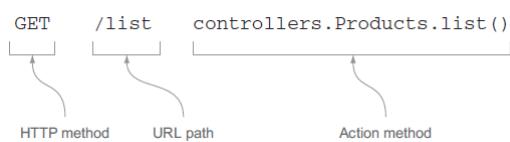
2.3 Play Framework

- 2 Play Framework[3] merupakan sebuah web *framework* berbasis bahasa Java dan Scala. Play Framework juga menggunakan *design pattern* Model-View-Controller (MVC) di mana *model* dan *controller* menggunakan bahasa Java sedangkan *view* menggunakan bahasa Scala dan HTML. Struktur aplikasi Play Framework dapat dilihat pada gambar 2.10.



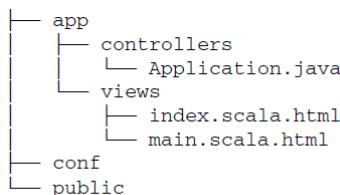
Gambar 2.10: Struktur Aplikasi Play Framework

- 6 Dalam direktori **conf**, terdapat file **routes**. Melalui **routes**, rute aplikasi dapat ditentukan dengan memetakan URL ke kode aplikasi. Setiap *route* memiliki tiga bagian yaitu *HTTP method*, *URL path*, dan *action method*. *HTTP method* merupakan metode pengiriman *HTTP*. *URL path* merupakan URL untuk mengakses halaman. *Action method* merupakan *method* yang menangani permintaan metode pengiriman *HTTP*. Sebagai contoh pada gambar 2.11, setiap permintaan GET pada URL /list akan ditangani oleh *method* `list()` milik kelas `Products` yang terdapat pada *package* `controllers`.



Gambar 2.11: Contoh Komponen Route[3]

1 Direktori `app` merupakan sumber dari kode program seperti file Java dan *view*. Saat
 2 pertama kali proyek Play Framework dibuat, direktori `app` berisi file-file seperti pada gam-
 3 bar 2.12. Dalam folder `controllers`, terdapat file `Application.java` yang berisi kode Java
 4 untuk menghasilkan halaman web. Kelas yang menangani permintaan HTTP dan mengem-
 5 balikan hasil HTTP disebut kelas *controller*. Kelas *controller* merupakan kelas yang memiliki
 6 *action method*. Setiap *action method* memiliki tipe kembalian *Result* yang merepresenta-
 7 sikan *view*. *Action method* kan berhubungan dengan *view* setelah didefinisikan di `routes`.
 8 *Controller* dapat mengirimkan *parameter* pada *view* melalui kembalian dari *action method*.
 9 Sebagai contoh pada gambar 2.13, `method home()` mengembalikan *view* `home` yang berada
 10 pada *package views* dengan mengirim *parameter* “nama”.



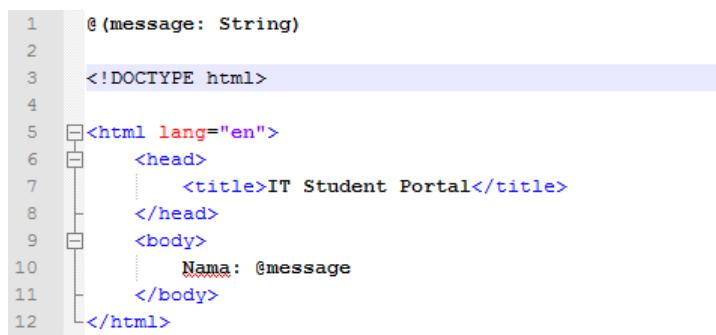
Gambar 2.12: Direktori app yang Dibangkitkan Play Framework[3]

```

public Result home() {
    String nama = scrap.getLoggedMahasiswa().getNama();
    return ok(views.html.home.render(nama));
}
  
```

Gambar 2.13: Contoh Method pada Kelas Controller

11 Dalam folder `views` terdapat dua file yaitu `index.scala.html` dan `main.scala.html`
 12 yang berfungsi untuk mendefinisikan halaman HTML. Setiap konten yang dihasilkan pada
 13 server dan dikirim ke klien dalam *body* HTTP, seperti halaman HTML, disebut *view*. *View*
 14 dapat menerima *parameter* dari *controller* menggunakan bahasa Scala. Sebagai contoh
 15 pada gambar 2.14, pada baris pertama “message” mendefinisikan nama *parameter* yang
 16 diterima dengan tipe *String*. Tipe yang diterima *view* harus sama dengan tipe yang dikirim
 17 *controller* begitu pula banyak *parameter*-nya. Baris ke-10 menampilkan “message” pada
 18 halaman HTML. Tanda “@” menandakan penggunaan bahasa Scala pada *view*. Folder-folder
 19 yang terdapat dalam direktori `app` akan menjadi *package* dalam kode Java.



Gambar 2.14: Penerimaan *Parameter* pada *View*

1 Direktori `public` berisi sumber yang dapat diakses secara langsung sebagai aset publik.
 2 Biasanya aset publik mendukung file selain aplikasi yang dibuat seperti gambar, *stylesheet*,
 3 Javascript, dan halaman HTML statis. Aset publik tidak dihasilkan oleh aplikasi melainkan
 4 diatur secara langsung oleh pembuat program.

5 Dalam Play Framework, objek yang disimpan pada *session* memiliki masa hidup yaitu
 6 selama *browser* dibuka. *Session* tidak disimpan di server melainkan ditambahkan ke setiap
 7 permintaan HTTP berikutnya menggunakan mekanisme *cookie*. Ukuran data *session* sangat
 8 terbatas yaitu hingga 4 KB sehingga hanya dapat menyimpan *String*. Pada controller,
 9 *session* dapat disimpan dengan *method*:

- 10 • `public static void session(String key, String value)`

11 **Parameter:**

- 12 – **key** kunci *session*.
 13 – **value** nilai *session*.

14 Sedangkan nilai *session* dapat diperoleh menggunakan *method*:

- 15 • `public static String session(String key)`

16 **Parameter:**

- 17 – **key** kunci *session*.

18 **Kembalian:** nilai *session*.

19 2.4 SIA Models

20 SIA Models merupakan kelas-kelas dalam bahasa Java yang merepresentasikan Sistem In-
 21 formasi Akademik UNPAR[6]. Kelas-kelas yang dimiliki SIA Models terbagi ke dalam tiga
 22 package antara lain:

- 23 1. *Package* `id.ac.unpar.siamodels`

24 *Package* ini memiliki kelas-kelas sebagai berikut:

- 25 (a) Mahasiswa

26 Kelas ini merepresentasikan mahasiswa. *Method-method* yang dimiliki kelas ini
 27 adalah sebagai berikut:

- 1 ● **public Mahasiswa(String npm)**
- 2 Merupakan *constructor* dari kelas Mahasiswa.
- 3 **Parameter:**
- 4 – **npm** nomor pokok mahasiswa.
- 5 ● **public String getNama()**
- 6 Berfungsi untuk mendapatkan nama mahasiswa.
- 7 **Kembalian:** nama mahasiswa.
- 8 ● **public void setNama(String nama)**
- 9 Berfungsi untuk mengubah nama mahasiswa.
- 10 **Parameter:**
- 11 – **nama** nama mahasiswa.
- 12 ● **public String getNpm()**
- 13 Berfungsi untuk mendapatkan nomor pokok mahasiswa.
- 14 **Kembalian:** nomor pokok mahasiswa.
- 15 ● **public String getEmailAddress()**
- 16 Berfungsi untuk mendapatkan *email* mahasiswa.
- 17 **Kembalian:** *email* mahasiswa.
- 18 ● **public List<Nilai> getRiwayatNilai()**
- 19 Berfungsi untuk mendapatkan riwayat nilai mahasiswa.
- 20 **Kembalian:** riwayat nilai mahasiswa dalam List.
- 21 ● **public double calculateIPKLulus()**
- 22 Menghitung IPK mahasiswa sampai saat ini, dengan aturan kuliah yang
- 23 tidak lulus tidak dihitung dan jika pengambilan beberapa kali, diambil nilai
- 24 terbaik. Sebelum memanggil *method* ini, *getRiwayatNilai()* harus sudah
- 25 mengandung nilai per mata kuliah.
- 26 **Kembalian:** IPK lulus.
- 27 ● **public double calculateIPS()**
- 28 Menghitung IPS semester terakhir sampai saat ini, dengan aturan kuliah
- 29 yang tidak lulus dihitung. Sebelum memanggil *method* ini, *getRiwayatNilai()*
- 30 harus sudah mengandung nilai per mata kuliah.
- 31 **Kembalian:** nilai IPS sampai saat ini.
- 32 ● **public int calculateSKSLulus()**
- 33 Menghitung jumlah SKS lulus mahasiswa saat ini. Sebelum memanggil *me-*
- 34 *thod* ini, *getRiwayatNilai()* harus sudah mengandung nilai per mata kuliah.
- 35 **Kembalian:** SKS lulus.
- 36 ● **public boolean hasLulusKuliah(String kodeMataKuliah)**
- 37 Memeriksa apakah mahasiswa ini sudah lulus mata kuliah tertentu. Sebelum
- 38 memanggil *method* ini, *getRiwayatNilai()* harus sudah mengandung nilai
- 39 per mata kuliah.
- 40 **Parameter:**
- 41 – **kodeMataKuliah** kode mata kuliah yang ingin diperiksa kelulusannya.
- 42 **Kembalian:** **true** jika sudah pernah mengambil dan lulus, **false** jika belum.
- 43 ● **public boolean hasTempuhKuliah(String kodeMataKuliah)**

1 Memeriksa apakah mahasiswa ini sudah pernah menempuh mata kuliah ter-
2 tentu. Sebelum memanggil *method* ini, *getRiwayatNilai()* harus sudah
3 mengandung nilai per mata kuliah.

4 **Parameter:**

- 5 – **kodeMataKuliah** kode mata kuliah yang ingin diperiksa kelulusannya.

6 **Kembalian:** *true* jika sudah pernah mengambil, *false* jika belum.

- 7 • **public int getTahunAngkatan()**

8 Mendapatkan tahun angkatan mahasiswa ini berdasarkan NPM-nya.

9 **Kembalian:** tahun angkatan.

10 (b) Nilai

11 Kelas ini merepresentasikan nilai yang ada pada riwayat nilai mahasiswa. *Method-*
12 *method* yang dimiliki kelas ini adalah sebagai berikut:

- 13 • **public Nilai(int tahunAjaran, int semester, MataKuliah mataKu-
14 liah, Character kelas, Double nilaiART, Double nilaiUTS, Double
15 nilaiUAS, Character nilaiAkhir)**

16 Merupakan *constructor* dari kelas Nilai.

17 **Parameter:**

- 18 – **tahunAjaran** tahun ajaran kuliah ini diambil.

- 19 – **semester** semester kuliah ini diambil.

- 20 – **mataKuliah** mata kuliah yang diambil.

- 21 – **kelas** kelas kuliah.

- 22 – **nilaiART** nilai ART.

- 23 – **nilaiUTS** nilai UTS.

- 24 – **nilaiUAS** nilai UAS.

- 25 – **nilaiAkhir** nilai akhir.

- 26 • **public MataKuliah getMataKuliah()**

27 Mendapatkan mata kuliah yang diambil.

28 **Kembalian:** mata kuliah.

- 29 • **public Character getKelas()**

30 Mendapatkan kelas kuliah.

31 **Kembalian:** kelas kuliah.

- 32 • **public Double getNilaiART()**

33 Mendapatkan nilai ART.

34 **Kembalian:** nilai ART.

- 35 • **public Double getNilaiUTS()**

36 Mendapatkan nilai UTS.

37 **Kembalian:** nilai UTS.

- 38 • **public Double getNilaiUAS()**

39 Mendapatkan nilai UAS.

40 **Kembalian:** nilai UAS.

- 41 • **public Double getNilaikhir()**

42 Mendapatkan nilai akhir dalam bentuk angka.

43 **Kembalian:** nilai akhir dalam huruf atau *null* jika tidak ada.

- **public Double getAngkaAkhir()**
Mengembalikan nilai akhir dalam bentuk huruf (A, B, C, D, ...).
Kembalian: nilai akhir dalam angka, atau **null** jika `getNilaiAkhir()` mengembalikan **null**.
 - **public int getTahunAjaran()**
Mendapatkan tahun ajaran saat pengambilan mata kuliah.
Kembalian: tahun ajaran saat pengambilan mata kuliah.
 - **public int getTahunSemester()**
Mendapatkan semester pengambilan mata kuliah.
Kembalian: semester pengambilan mata kuliah.
- (c) **ChronologicalComparator**
Pembanding antara satu nilai dengan nilai lainnya, secara kronologis waktu pengambilan. *Method* yang dimiliki kelas ini adalah sebagai berikut:
- **public int compare(Nilai o1, Nilai o2)**
Berfungsi untuk membandingkan nilai.
Parameter:
 - **o1** nilai pertama yang akan dibandingkan.
 - **o2** nilai kedua yang akan dibandingkan.
Kembalian: hasil perbandingan.
- (d) **MataKuliah**
Kelas ini merepresentasikan sebuah mata kuliah. *Method-method* yang dimiliki kelas ini adalah sebagai berikut:
- **protected MataKuliah(String kode, int sks, String nama)**
Merupakan *constructor* dari kelas MataKuliah.
Parameter:
 - **kode** kode mata kuliah.
 - **sks** bobot SKS mata kuliah.
 - **nama** nama mata kuliah.
 - **public String getKode()**
Mendapatkan kode mata kuliah.
Kembalian: kode mata kuliah.
 - **public int getSKS()**
Mendapatkan bobot SKS mata kuliah.
Kembalian: bobot SKS mata kuliah.
 - **public String getNama()**
Mendapatkan nama mata kuliah.
Kembalian: nama mata kuliah.
 - **public static MataKuliah createMataKuliah(String kode, int sks, String nama)**
Mendapatkan atau membuat mata kuliah baru.
Parameter:
 - **kode** kode mata kuliah.

1 – **sks** bobot SKS mata kuliah.

2 – **nama** nama mata kuliah.

3 **Kembalian:** objek mata kuliah.

4 • **public static MataKuliah getMataKuliah(String kode)**

5 Mendapatkan mata kuliah.

6 **Parameter:**

7 – **kode** kode mata kuliah.

8 **Kembalian:** mata kuliah sesuai kode.

9 (e) Semester

10 Kelas ini menyimpan konstanta untuk semester-semester di UNPAR. Nilai konstanta harus sesuai urutan kronologis dalam satu tahun ajaran. *Method* yang dimiliki kelas ini adalah sebagai berikut:

13 • **public static final int fromString(String text)**

14 Berfungsi untuk mengubah semester dari bentuk teks ke konstanta.

15 **Parameter:**

16 – **text** semester dalam bentuk teks (GANJIL, GENAP, PENDEK).

17 **Kembalian:** konstanta semester.

18 2. *Package id.ac.unpar.siamodels.matakuliah.interfaces*

19 *Package* ini memiliki beberapa *interface* antara lain:

20 (a) HasPrasyarat

21 Mendefinisikan kelas-kelas yang memiliki prasyarat, terkustomisasi untuk seorang mahasiswa. *Method* yang dimiliki *interface* ini adalah sebagai berikut:

23 • **public boolean checkPrasyarat(Mahasiswa mahasiswa, List<String> reasonsContainer)**

25 Memeriksa prasyarat-prasyarat dari kuliah, spesifik untuk mahasiswa yang dituju. Jika ada pesan-pesan khusus, akan ditambahkan pada parameter reasonsContainer.

28 **Parameter:**

29 – **mahasiswa** prasyarat kuliah akan diperiksa spesifik pada mahasiswa ini.

30 – **reasonsContainer** jika pesan-pesan terkait prasyarat akan ditambahkan di sini.

32 **Kembalian:** `true` jika seluruh prasyarat dipenuhi, `false` jika tidak.

33 (b) Pilihan

34 Mendefinisikan kelas-kelas yang merupakan mata kuliah pilihan.

35 (c) PilihanWajib

36 Mendefinisikan kelas-kelas yang merupakan mata kuliah pilihan wajib.

37 (d) Wajib

38 Mendefinisikan kelas-kelas yang merupakan mata kuliah wajib.

39 3. *Package id.ac.unpar.siamodels.matakuliah*

40 *Package* ini berisi kelas-kelas yang merepresentasikan mata kuliah yang terdapat pada

- ¹ Program Studi Teknik Informatika UNPAR. Rincian dari kelas-kelas pada package ini
² dapat dilihat pada tabel 2.1.

Kelas	Implements	Kelas	Implements
AIF101	-	AIF342	HasPrasyarat
AIF102	HasPrasyarat, Wajib	AIF344	HasPrasyarat
AIF103	-	AIF360	HasPrasyarat
AIF105	-	AIF362	HasPrasyarat, Pilihan
AIF200	-	AIF401	HasPrasyarat, Wajib
AIF201	HasPrasyarat, Wajib	AIF402	HasPrasyarat, Wajib
AIF202	HasPrasyarat, Wajib	AIF403	-
AIF203	HasPrasyarat, Wajib	AIF405	HasPrasyarat, Wajib
AIF204	HasPrasyarat	AIF438	HasPrasyarat, Pilihan
AIF205	HasPrasyarat, Wajib	AIF441	-
AIF206	HasPrasyarat	AIF445	HasPrasyarat
AIF208	HasPrasyarat	AIF453	HasPrasyarat, Pilihan
AIF301	HasPrasyarat, Wajib	AIF456	-
AIF302	HasPrasyarat, Wajib	AIF457	HasPrasyarat, Pilihan
AIF303	HasPrasyarat, Wajib	AIF458	HasPrasyarat
AIF304	HasPrasyarat	AIF461	HasPrasyarat
AIF305	HasPrasyarat, Wajib	AIF462	-
AIF306	HasPrasyarat	AIF469	HasPrasyarat, Pilihan
AIF311	HasPrasyarat, PilihanWajib	APS402	HasPrasyarat, Wajib
AIF312	HasPrasyarat	MKU001	-
AIF314	HasPrasyarat, PilihanWajib	MKU002	-
AIF315	HasPrasyarat, PilihanWajib	MKU003	-
AIF316	HasPrasyarat	MKU004	-
AIF317	HasPrasyarat, PilihanWajib	MKU008	-
AIF318	HasPrasyarat	MKU009	-
AIF332	HasPrasyarat	MKU010	-
AIF336	-	MKU011	-
AIF339	HasPrasyarat	MKU012	-
AIF341	-		

Tabel 2.1: Tabel Rincian Kelas pada Package `id.ac.unpar.siamodels.matakuliah`

³ 2.5 CSS Selector

- ⁴ CSS(*Cascading Style Sheets*) memungkinkan adanya perubahan terhadap teks dari elemen
⁵ HTML yang sudah didefinisikan^[9]. Seperti yang ditampilkan pada gambar 2.15, definisi
⁶ CSS memiliki dua komponen yaitu selector dan properti. CSS *selector* digunakan untuk
⁷ mendefinisikan elemen HTML sedangkan properti mendefinisikan atribut beserta nilai.

```
h1{ selector
    color: gray; —— properti
}
```

Gambar 2.15: Format Penulisan Definisi CSS

1 Beberapa jenis CSS *Selector* antara lain:

2 1. **Element Selector**, memilih *tag* html.

3 Contoh: `h1`

4 Keterangan: *selector* mendefinisikan elemen h1.

5 2. **Grouping Selector**, memilih beberapa *selector* sekaligus. Setiap *selector* dipisahkan
6 dengan “,”.

7 Contoh: `h1, h2, p`

8 Keterangan: *selector* mendefinisikan elemen h1, h2, dan p.

9 3. **Universal Selector**, memilih seluruh elemen. *Selector* ditampilkan sebagai “*”.

10 Contoh: `*`

11 Keterangan: *selector* mendefinisikan seluruh elemen.

12 4. **Class Selector**, memilih kelas elemen. *Selector* ditampilkan sebagai “.” kemudian
13 diikuti nama kelas elemen.

14 Contoh: `.top`

15 Keterangan: *selector* mendefinisikan elemen dengan kelas “top”.

16 5. **ID Selector**, memilih ID elemen. *Selector* ditampilkan sebagai “#” kemudian diikuti
17 ID elemen.

18 Contoh: `#top`

19 Keterangan: *selector* mendefinisikan elemen dengan ID “top”.

20 6. **Attribute Selector**, akan dijelaskan dua *attribute selector* yaitu:

21 • **Simple Attribute**, memilih atribut elemen. *Selector* ditampilkan sebagai nama
22 atribut kemudian diapit dengan kurung siku. Contoh: `[name]`

23 Keterangan: *selector* mendefinisikan elemen dengan atribut “name”.

24 • **Exact Value Attribute**, memilih atribut elemen dengan nilai tertentu. *Selector*
25 ditampilkan sebagai definisi atribut kemudian diapit dengan kurung siku. Contoh:
26 `[name=Joe]`

27 Keterangan: *selector* mendefinisikan elemen dengan atribut “name” yang memiliki
28 nilai “Joe”.

29 7. **Descendant Selector**, memilih *child* elemen yang merupakan keturunan *parent* ter-
30 tentu. *Selector* ditampilkan dengan mendefinisikan *parent* kemudian diikuti oleh *child*
31 dipisahkan dengan spasi.

32 Contoh: `p .top`

33 Keterangan: *selector* mendefinisikan elemen dengan kelas “top” yang merupakan *child*
34 dari elemen p.

1

BAB 3

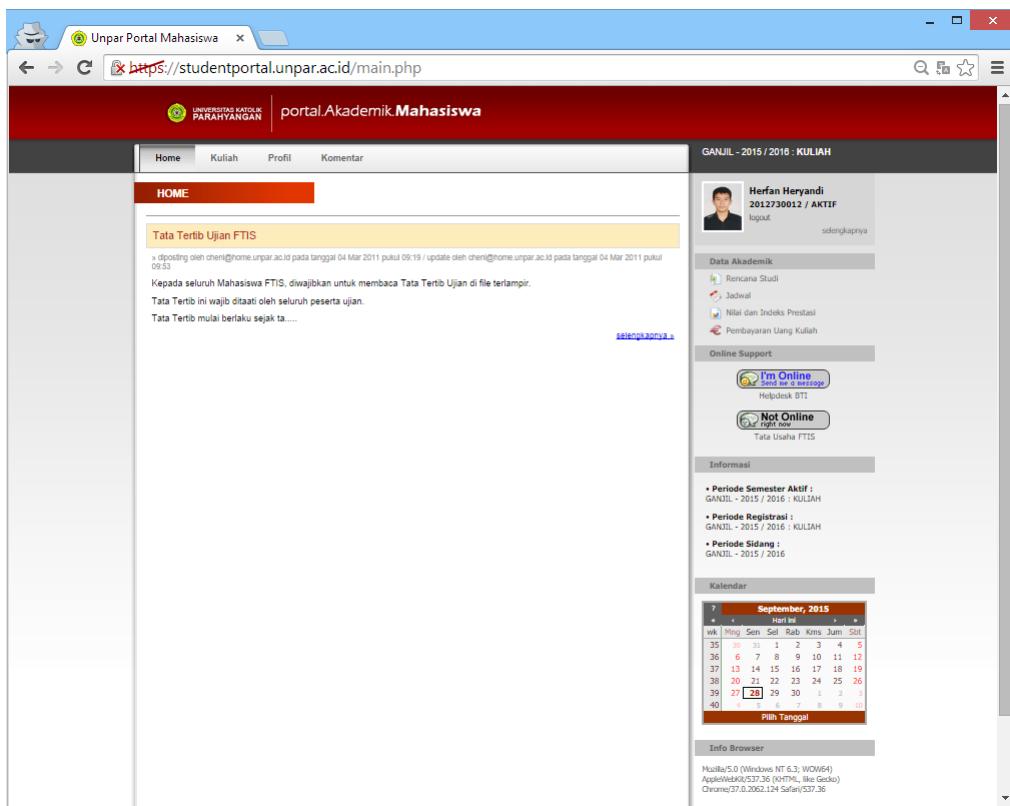
2

ANALISIS

3

3.1 Analisis Portal Akademik Mahasiswa

- 4 Portal Akademik Mahasiswa merupakan sebuah situs jaringan yang diperuntukan bagi mahasiswa dalam rangka mendapatkan informasi kegiatan akademik[4]. Mahasiswa dapat mengakses Portal Akademik Mahasiswa melalui URL <https://studentportal.unpar.ac.id/>.
- 5 Untuk mengakses Portal Akademik Mahasiswa, mahasiswa harus *login* menggunakan akun email *student*. Halaman *login* Student Portal UNPAR terintegrasi dengan CAS (*Central Authentication Service*) UNPAR¹.



Gambar 3.1: Halaman Utama Portal Akademik Mahasiswa

10 Pada halaman utama Portal Akademik Mahasiswa (gambar 3.1), terdapat beberapa
11 bagian yaitu:

12 1. Menu Atas

13 Menu ini berfungsi sebagai menu pendukung yang terdiri dari :

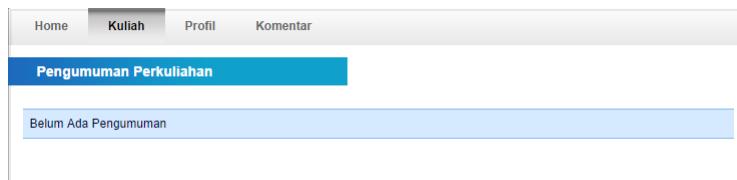
¹<https://cas.unpar.ac.id>

- 1 ● **Home**, menampilkan informasi atau pengumuman yang dikeluarkan oleh fakultas masing-masing (Gambar 3.2).



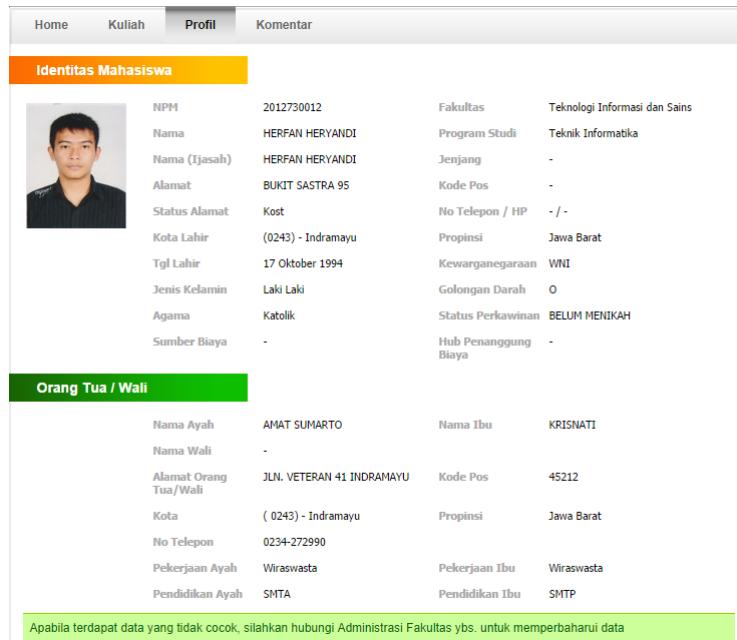
Gambar 3.2: Menu Atas Home

- 3 ● **Kuliah**, menampilkan pengumuman per mata kuliah sesuai dengan mata kuliah dan kelas yang diambil oleh masing-masing mahasiswa (Gambar 3.3).



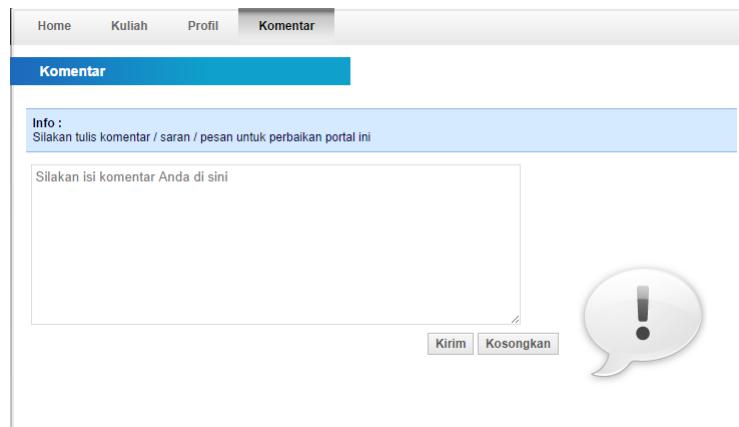
Gambar 3.3: Menu Atas Kuliah

- 5 ● **Profil**, berisi tentang data diri masing-masing mahasiswa (Gambar 3.4).



Gambar 3.4: Menu Atas Profil

- 6 ● **Komentar**, berisi komentar, saran, dan kritik dari mahasiswa (Gambar 3.5).



Gambar 3.5: Menu Atas Komentar

1 2. Identitas Portal

2 Bagian ini menampilkan identitas pengguna portal. Tampilan identitas ini dapat di-tampilkan lengkap dengan melakukan klik pada link “selengkapnya” atau ditampilkan minimal dengan klik link “tutup”. Identitas yang ditampilkan adalah nama, Nomor Pokok Mahasiswa (NPM), status keaktifan, pas foto, email, dosen wali, program studi, dan fakultas seperti yang terlihat pada gambar 3.6.



Gambar 3.6: Identitas Portal

7 3. Menu Utama

8 Bagian ini memuat fitur utama Portal Akademik Mahasiswa mengenai data akademik (gambar 3.7) yang terdiri dari:



Gambar 3.7: Menu Utama

10 • Rencana Studi

11 Menu Rencana Studi terdiri dari submenu:

- 12 – Registrasi (FRS/PRS)

- 1 Digunakan sebagai formulir pengisian rencana studi awal (FRS) dan perubahan rencana studi (PRS) (Gambar 3.8).

No.	Kode MK	Nama MK	Pada Pengampuan	Pilih FRS	Pilih PRS
1	EEP101-3	Pengantar Ekonomi Mikro		<input checked="" type="checkbox"/>	-
2	EMS101-3	Pengantar Bisnis		<input checked="" type="checkbox"/>	-
3	MHU001-2	Pendidikan Pancasila		<input checked="" type="checkbox"/>	-
4	MHU002-2	Pendidikan Kinerjanegeoran		<input checked="" type="checkbox"/>	-
Mata Kuliah Pilihan					
5	EAR205-0	Responsi Akuntansi Keuangan Lanjut I	Akuntansi	<input checked="" type="checkbox"/>	-
6	EMS205-3	Perilaku Organisasi	Manajemen	<input checked="" type="checkbox"/>	-
7	EMS305-3	Statistika Non Parametrik	Manajemen	<input checked="" type="checkbox"/>	-
8	EMS306-3	Kewirausahaan	Manajemen	<input checked="" type="checkbox"/>	-
		Total Sks	19	0	
		Total Mata Kuliah	8	0	
		Total Sks Akhir	19	sks	
		Total Mata Kuliah Akhir	8	mata kuliah	

[Simpan](#) [Kembali](#)

Gambar 3.8: Tampilan Registrasi FRS/PRS

3 – Kartu Rencana Studi

4 Menampilkan informasi mata kuliah yang telah diambil melalui submenu
5 Registrasi (Gambar 3.9). Kartu Rencana Studi juga dapat dicetak melalui
6 submenu ini.

No.	Kode MK	Nama Matakuliah	sks	Kelas	SAPR
1	AIF401	Skripsi 1	4	A	
2	AIF402	Skripsi 2	6	A	
3	AIF439	Kerja Praktek 2	3	A	
4	AIF441	Administrasi Jaringan Komputer 3	3	A	P
5	AIF457	Kewirausahaan Berbasis Teknologi	3	A	
6	AIF469	Layanan Berbasis Web	3	A	
Total sks			22		

* Anda sudah pernah melakukan registrasi pada tanggal : 6 Juli 2015

[Cetak KRS](#)

Keterangan :
S : Studio
A : Asistensi
P : Praktikum
R : Responsi

Gambar 3.9: Tampilan Kartu Rencana Studi[4]

7 – Pindah Kelas MKU

8 Mahasiswa dapat memilih kelas yang masih tersedia di kolom Jadwal Baru
9 dan menekan tombol “Simpan” untuk setiap kelas yang diubah (Gambar
10 3.10).

Ubah Jadwal MKU Anda :
GANJIL - 2010 / 2011

Matakuliah	Jadwal Sekarang	Jadwal Baru
MKU001 - Pendidikan Pancasila	Kelas : C Jadwal : Kamis, 9.00 - 11.00 Ruang : 10118 Dosen :	Kelas : D (0/50) <input type="button" value="▼"/> <input type="button" value="Simpan"/> Jadwal : Kamis, 11.00 - 13.00 Ruang : 10118 Dosen : Kapasitas : 50 Peserta :
MKU002 - Pendidikan Kewarganegaraan	Kelas : A Jadwal : Senin, 8.00 - 10.00 Ruang : 2131 Dosen :	Kelas : Tidak Berubah <input type="button" value="▼"/> <input type="button" value="Simpan"/> Jadwal : - Ruang : - Dosen : - Kapasitas : - Peserta : -

Gambar 3.10: Tampilan Pindah Kelas MKU[4]

1 ● **Jadwal**

2 Menu Jadwal terdiri dari submenu:

- 3 – Kuliah, UTS, dan UAS

4 Submenu ini berisi tentang jadwal kuliah, UTS dan UAS yang dapat disusun per semester (Gambar 3.11).

Jadwal Kuliah / UTS / UAS

GANJIL - 2015 / 2016

Jadwal Kuliah

KULIAH : GANJIL - 2015 / 2016

No	Kode	Nama Mata Kuliah	skls	Kelas	Nama Dosen	Temu	Hari	Waktu	Ruang
1	AIF401	Skripsi 1	4	A	Marikha Tri Adithia, S.Si., M.Sc., PDEng.	1	Selasa	07.00-09.00	10316
2	AIF402	Skripsi 2	6	A	Marikha Tri Adithia, S.Si., M.Sc., PDEng.	1	Rabu	10.00-12.00	09122
3	AIF439	Keja Praktek 2	3	A	-		1	00.00-00.00	00000
4	AIF441	Administrasi Jaringan Komputer 3	3	A	Chandra Wijaya, S.T., M.T.	1	Kamis	07.00-10.00	09016
			3	A	Billy Susanto Panca, S.T.	2	Senin	15.00-17.00	09016
5	AIF457	Kewirausahaan Berbasis Teknologi	3	A	Dr. Veronica Sri Moertini, Ir., M.T.	1	Jumat	08.00-11.00	10317
6	AIF469	Layanan Berbasis Web	3	A	Pascal Alfadian, S.Kom., M.Com.	1	Kamis	13.00-16.00	09122
					Total sks	22	sks		

Jadwal UTS

UTS : GANJIL - 2015 / 2016

No	Kode	Nama Mata Kuliah	skls	Kelas	Tanggal Ujian	Waktu	Ruang	Kursi
1	AIF441	Administrasi Jaringan Komputer 3	3	A	13 OCT 15	08.00-10.00	09016	0007
2	AIF469	Layanan Berbasis Web	3	A	16 OCT 15	13.30-15.30	09120	0025

Jadwal UAS

Jadwal UAS untuk tahun akademik : GANJIL - 2015 / 2016 belum tersedia.

Gambar 3.11: Tampilan Jadwal Kuliah, UTS, dan UAS

6 – MKU

7 Submenu ini menampilkan seluruh jadwal Mata Kuliah Umum (MKU) yang memberikan informasi tentang kelas-kelas yang dibuka oleh Pusat Kajian

1 Humaniora (PKH) (Gambar 3.12).

Daftar Kelas & Jadwal MKU Semester Ini					
Silahkan klik Kode Mata Kuliah untuk melihat jadwal.					
MKU001 - Pendidikan Pancasila					
MKU002 - Pendidikan Kewarganegaraan					
MKU003 - Pendidikan Agama (Katolik)					
MKU004 - Pendidikan Agama (Fenomenologi)					
MKU008 - Etika					
MKU011 - Estetika					
MKU012 - Logika					
Kelas	Hari	Waktu	Ruang	Dosen	
1	Jumat	08.00 - 09.40	10119	Andreas Doweng Bolo, S.S., M.Hum.	
%	Jumat	10.00 - 11.40	10119	Bernardus Arto Tejo Sugiantoro, S.S., M.Hum.	
(Jumat	13.00 - 14.40	10119	Rudi Setiawan, S.Ag., M.M.	
)	Senin	07.00 - 08.40	10118	Yusuf Siswantara, S.S., M.Hum.	
1	Senin	15.00 - 16.40	10112	Yusuf Siswantara, S.S., M.Hum.	
2	Selasa	15.00 - 16.40	10112	Yusuf Siswantara, S.S., M.Hum.	
3	Rabu	10.00 - 11.40	10118	Yusuf Siswantara, S.S., M.Hum.	
4	Jumat	10.00 - 11.40	10116	Kristian Widya Wicaksono, S.Sos. M.Si.	
5	Senin	08.00 - 09.40	10116	Adelia, S.S., M.Si.	
6	Senin	10.00 - 11.40	10116	Adelia, S.S., M.Si.	
7	Senin	13.00 - 14.40	10116	Adelia, S.S., M.Si.	
8	Senin	15.00 - 16.40	10116	Adelia, S.S., M.Si.	
9	Selasa	08.00 - 09.40	10117	Adelia, S.S., M.Si.	
A	Kamis	10.00 - 12.00	06113	Dr. Stephanus Djunafan	
B	Selasa	10.00 - 11.40	10117	Adelia, S.S., M.Si.	
C	Selasa	13.00 - 14.40	10117	Adelia, S.S., M.Si.	
D	Selasa	09.00 - 10.40	10114	Aloysius Oscar Yasunari, S.S., M.M.	
E	Selasa	11.00 - 12.40	10114	Aloysius Oscar Yasunari, S.S., M.M.	

Gambar 3.12: Tampilan Jadwal MKU

2 – Seluruh Fakultas

3 Fitur ini memberikan informasi mengenai jadwal-jadwal yang ada di seluruh
4 fakultas (Gambar 3.13).

Jadwal Kuliah										
No	Kode	Nama Mata Kuliah	skS	Kelas	Nama Dosen	Temu	Hari	Waktu	Ruang	
1	AIF101	Pemrograman Berorientasi Objek	6	A	Husnul Hakim, S.Kom., M.T.	1	Senin	08.00-09.00	09122	
			6	A	Vania Natali, S.Kom.	2	Selasa	09.00-11.00	09018	
			6	A	Husnul Hakim, S.Kom., M.T.	3	Rabu	08.00-10.00	09018	
			6	A	Vania Natali, S.Kom.	4	Jumat	09.00-11.00	09018	
2	AIF101	Pemrograman Berorientasi Objek	6	A	Husnul Hakim, S.Kom., M.T.	1	Senin	08.00-10.00	09122	
			6	A	Vania Natali, S.Kom.	2	Selasa	09.00-11.00	09018	
			6	A	Husnul Hakim, S.Kom., M.T.	3	Rabu	08.00-10.00	09018	
			6	A	Vania Natali, S.Kom.	4	Jumat	09.00-11.00	09018	
3	AIF101	Pemrograman Berorientasi Objek	6	A	Husnul Hakim, S.Kom., M.T.	1	Senin	08.00-09.00	09122	
			6	A	Vania Natali, S.Kom.	2	Selasa	09.00-11.00	09018	
			6	A	Husnul Hakim, S.Kom., M.T.	3	Rabu	08.00-10.00	09018	
			6	A	Vania Natali, S.Kom.	4	Jumat	09.00-11.00	09018	
4	AIF101	Pemrograman Berorientasi Objek	6	A	Husnul Hakim, S.Kom., M.T.	1	Senin	08.00-10.00	09122	
			6	A	Vania Natali, S.Kom.	2	Selasa	09.00-11.00	09018	
			6	A	Husnul Hakim, S.Kom., M.T.	3	Rabu	08.00-10.00	09018	
			6	A	Vania Natali, S.Kom.	4	Jumat	09.00-11.00	09018	
5	AIF101	Pemrograman Berorientasi Objek	6	B	Vania Natali, S.Kom.	1	Senin	08.00-10.00	09121	
			6	B	Ferdian Reynaldi, S.T.	2	Selasa	09.00-11.00	09017	
			6	B	Vania Natali, S.Kom.	3	Rabu	08.00-10.00	09017	

Gambar 3.13: Tampilan Jadwal Seluruh Fakultas

1 ● **Nilai dan Indeks Prestasi**

2 Menu Nilai dan Indeks Prestasi terdiri dari submenu:

- 3 – Riwayat per Semester

4 Submenu ini menampilkan informasi nilai per semester. Mahasiswa dapat
5 melihat nilai sesuai dengan semester yang dipilih atau bisa memilih pilihan
6 “Seluruh Tahun Akademik” untuk melihat seluruh nilai berdasarkan semester
7 (Gambar 3.14).

Daftar Nilai Semester								GANJIL - 2015 / 2016	
No.	Kode MK	Nama Matakuliah	skls	Kelas	ART	UTS	UAS	AA	NA
1	AIF401	Skripsi 1 #	4	A	0	0	0	0	0
2	AIF402	Skripsi 2 #	6	A	0	0	0	0	0
3	AIF439	Kerja Praktek 2 #	3	A	0	0	0	0	0
4	AIF441	Administrasi Jaringan Komputer 3 #	3	A	0	0	0	0	0
5	AIF457	Kewirausahaan Berbasis Teknologi #	3	A	0	0	0	0	0
6	AIF469	Layanan Berbasis Web #	3	A	0	0	0	0	0
Total skls			22						

8 Keterangan :
9 \$: Nilai tidak dapat diihat karena status pembayaran belum lunas
10 # : Nilai belum tersedia
11 % : Nilai sedang dalam proses

Gambar 3.14: Tampilan Riwayat Per Semester

- 8 – Daftar Perkembangan Studi

9 Seluruh riwayat mata kuliah dan nilai yang pernah ditempuh ditampilkan di
10 submenu ini (Gambar 3.15). Pada bagian bawah halaman, terdapat statistik
11 nilai dan indeks prestasi (Gambar 3.16).

Daftar Perkembangan Studi							
Kode MK	Nama MK	Nilai	Tahun Sem	Kode MK	Nama MK	Nilai	Tahun Sem
SEMESTER 1							
AIF103-3	Matematika Diskrit	B	121	AIF104-4	Algoritma & Struktur Data	A	122
AIF105-3	Pengantar Informatika	B	121	AIF104-3	Logika Informatika	A	122
AIF191-3	Pemrograman Berorientasi Objek	B	121	AIF108-3	Sistem Dijital	A	122
MKU001-2	Pendidikan Pancasila	B	121	AMS190-3	Matematika Informatika	B	122
MKU008-2	Eтика	B	121	MKU009-2	Bahasa Indonesia	A	141
MKU010-2	Bahasa Inggris	A	122	MKU011-2	Estatika	B	122
SEMESTER 3							
AIF201-4	Analisis & Desain Berorientasi Objek	A	131	AIF202-4	Desain & Analisis Algoritma	B	132
AIF203-4	Struktur Diskrit	A	131	AIF204-4	Manajemen Informasi & Basis Data	B	132
AIF205-3	Arsitektur & Organisasi Komputer	A	131	AIF206-4	Sistem Operasi	B	132
AMS200-3	Probabilitas & Statistika	A	131	AIF208-4	Rekayasa Perangkat Lunak	B	132
MKU003-2	Pendidikan Agama (katolik)	B	131	AIF210-2	Interaksi Manusia Komputer	B	132
MKU012-2	Logika	A	122				
SEMESTER 5							
AIF301-3	Pengantar Sistem Cerdas	C	141	AIF302-2	Penulisan Ilmiah	B	142
AIF303-3	Pengantar Sistem Informasi	A	141	AIF305-5	Proyek Informatika	A	142
AIF305-4	Jaringan Komputer	A	141				
MKU002-2	Pendidikan Kewarganegaraan	B	122				
SEMESTER 7							
AIF403-2	Komputer & Masyarakat	B	141	APS402-2	Etika Profesi	A	142
Mata Kuliah Pilihan							
AMS191-4	Kalkulus	B	121				
AIF311-2	Pemrograman Fungsional	A	141				
AIF313-2	Grafika Komputer	B	131				
AIF315-2	Pemrograman Berbasis Web	A	141				
AIF341-3	Administrasi Jaringan Komputer 1	A	141				
AIF312-2	Keamanan Informasi	D	142				
AIF318-2	Pemrograman Aplikasi Bergerak	A	142				
AIF342-3	Administrasi Jaringan Komputer 2	A	142				
AIF390-3	Pemrograman Berbasis Web Lanjut	B	142				
AIF445-3	Metode Numerik	B	132				
AIF455-3	Sistem Pendukung Keputusan	A	141				
AIF461-2	Pencarian & Temu Kembali Informasi	A	142				
AIF480-3	Pengolahan Citra	A	131				
APS302-2	Dunia Digital Dan Sains	A	142				

Gambar 3.15: Tampilan Daftar Perkembangan Studi

Kode Semester :						
1 = Ganjil, 2 = Genap, 4 = Padat, 6 = Transfer						
Ket. Mt Kuliah :						
• [M] = Mk. Kendali Mutu,						
• [X] = Mk. Disisihkan,						
• [<] = Mk. yang diambil semester (2015-1)						
Nilai Akhir	A	B	C	D	E	K
Jml. Mata Kuliah	23	20	1	1		
Jml. sks	67	56	3	2		
Jml. Mk K. Mutu						
Jml. sks K. Mutu						
No	Skor	Tanggal				
1	544	30 Maret 2015				

IP Lulus (2014-2)	(128 sks)	: 3.47
IP N. Terbaik (2014-2)	(128 sks)	: 3.47
IPS (2014-2)	(24 sks)	: 3.54
IFT-1 (2013-2)	(30)	: 4.00 sks lulus : 80

Jumlah sks	
Ditempuh	: 128 sks
Lulus Wajib	: 92 sks
Lulus Pilihan	: 36 sks
Lulus Wajib Peminatan	: 0 sks
Lulus Pilihan Peminatan	: 0 sks
Total Lulus	: 128 sks

Ditempuh Semester ini (2014-2)	: 24 sks
Dijenjikan untuk semester yad.(2015-2)	: 24 sks

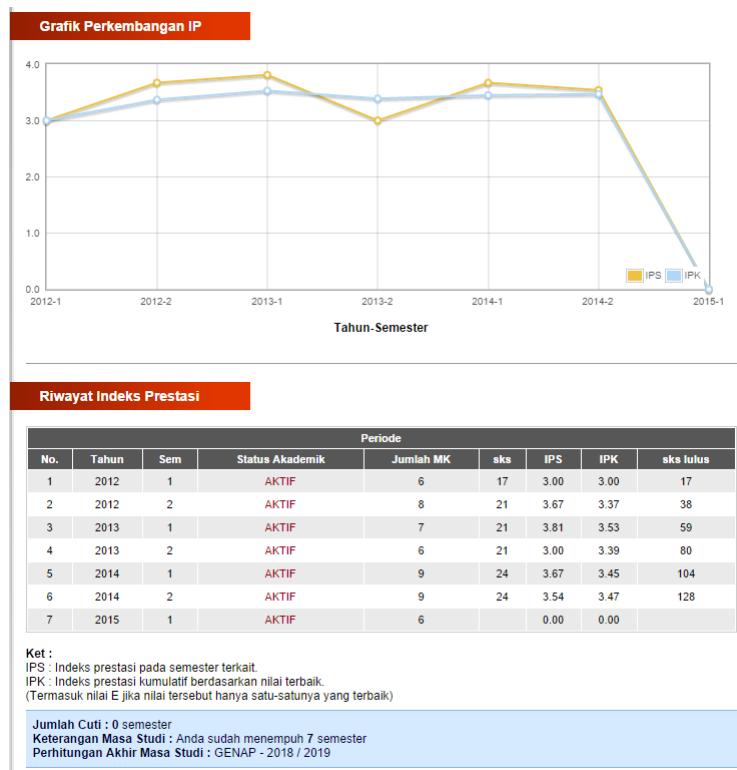
Cuti studi : 0 semester
Akhir Masa Studi : Semester GENAP - 2018 / 2019

Syarat Kelulusan :
Lulus min. 144 sks terdiri dari : Mk. Wajib + Mk. Pilihan
I.P. Lulus minimum : 2.00

Gambar 3.16: Tampilan Statistik Nilai dan IP

– Riwayat Indeks Prestasi

Menampilkan daftar riwayat indeks prestasi semester dan kumulatif setiap semester. Tampilan ini juga dilengkapi dengan grafik perkembangan (Gambar 3.17).



Gambar 3.17: Tampilan Riwayat Indeks Prestasi

– TOEFL

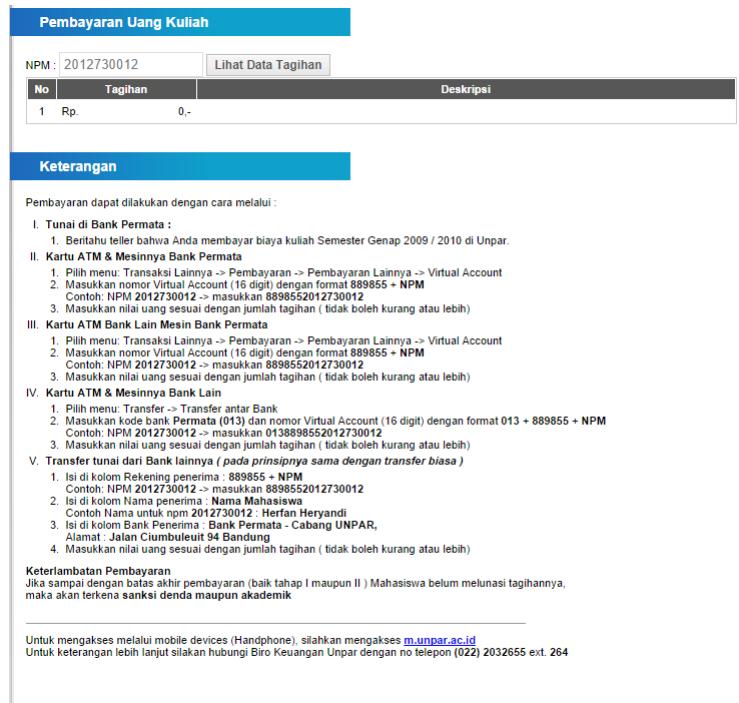
Menampilkan daftar riwayat skor *Test of English as Foreign Language* (TOEFL) yang pernah ditempuh (Gambar 3.18). Mahasiswa diwajibkan untuk menempuh TOEFL dengan skor minimal 500.



Gambar 3.18: Tampilan TOEFL

- Pembayaran Uang Kuliah

2 Menu ini berfungsi untuk melihat data tagihan pembayaran uang kuliah serta
3 cara-cara pembayarannya (Gambar 3.19).



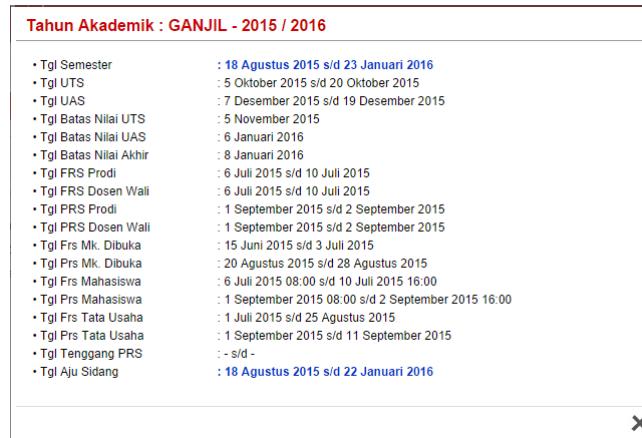
Gambar 3.19: Tampilan Pembayaran Uang Kuliah

4. Informasi

5 Bagian ini menampilkan informasi tentang periode-periode yang sedang aktif (Gambar
6 3.20). Sebagai contoh jika “Periode Registrasi” diklik maka akan muncul *pop up* seperti
7 pada gambar 3.21.



Gambar 3.20: Tampilan Informasi



Gambar 3.21: Tampilan *Pop Up* Informasi

1 5. Kalender

- 2 Bagian ini menampilkan kalender masehi (Gambar 3.22).



Gambar 3.22: Tampilan Kalender

3 6. Info Browser

- 4 Bagian ini menampilkan informasi tentang internet browser yang digunakan pada saat
5 membuka Portal Akademik Mahasiswa (Gambar 3.23).



Gambar 3.23: Tampilan Info Browser

3.2 Analisis Kebutuhan IT Student Portal

Dalam menganalisis kebutuhan IT Student Portal, penulis melakukan wawancara dengan 18 mahasiswa Program Studi Teknik Informatika UNPAR. Kriteria dari 18 mahasiswa tersebut yaitu sembilan mahasiswa angkatan 2012, delapan mahasiswa angkatan 2013, dan satu mahasiswa angkatan 2014. Setelah melakukan wawancara, penulis memperoleh fitur-fitur yang diinginkan mahasiswa antara lain:

1. Prasyarat mata kuliah

Mahasiswa bisa memeriksa prasyarat mata kuliah saat FRS sehingga tidak terjadi kesalahan pengambilan mata kuliah. Prasyarat mata kuliah yang ditampilkan di Portal Akademik Mahasiswa kurang akurat. Selain itu, dari 18 mahasiswa yang diwawancara, hanya ada satu mahasiswa yang mengetahui bahwa Portal Akademik Mahasiswa memiliki fitur prasyarat. Prasyarat mata kuliah untuk Program Studi Teknik Informatika juga tersedia di <http://tinyurl.com/lionov>, namun mahasiswa merasa kurang praktis karena harus memeriksa secara manual. Mahasiswa menginginkan agar fitur ini bisa dibuat untuk mempermudah FRS.

2. Ringkasan data akademik

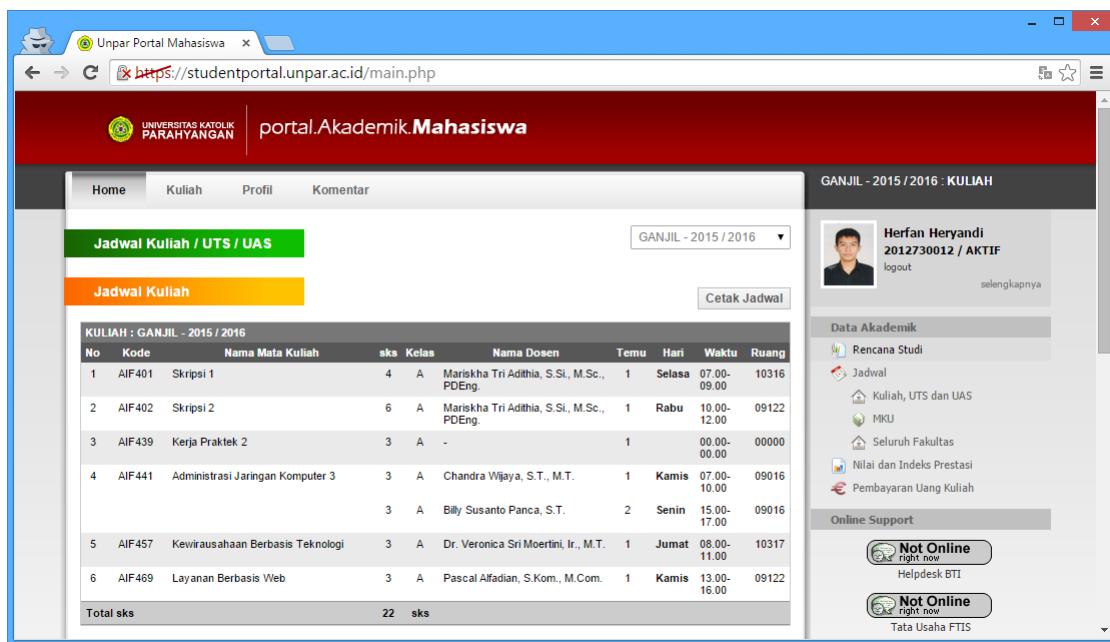
Ringkasan data akademik menampilkan data mengenai mata kuliah wajib, pilihan, dan pilihan wajib yang sudah lulus, mata kuliah wajib yang belum lulus, dan sisa SKS untuk mencapai kelulusan. Mahasiswa menginginkan fitur ini dibuat untuk membantu mahasiswa dalam mengatur perkuliahananya.

3. Perubahan IPS dan IPK berdasarkan riwayat nilai

Dalam Portal Akademik Mahasiswa, nilai pertama kali muncul dalam riwayat nilai. Riwayat IP tidak berubah secara otomatis saat seluruh nilai di riwayat nilai sudah muncul. Mahasiswa menginginkan agar IPS dan IPK dapat berubah secara otomatis saat nilai muncul.

4. Jadwal kuliah yang tersusun

Tampilan jadwal kuliah dalam Portal Akademik Mahasiswa tidak terurut berdasarkan hari seperti pada gambar 3.24 sehingga perlu direkapitulasi lagi. Mahasiswa menginginkan agar tampilan jadwal tersusun dan dalam bentuk seperti gambar 3.25.



Gambar 3.24: Tampilan Jadwal pada Portal Akademik Mahasiswa

	SENIN	SELASA	RABU	KAMIS	JUMAT	SABTU
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						

Gambar 3.25: Tampilan Jadwal yang Diinginkan Mahasiswa

5. Kalender akademik

Kalender akademik merupakan salah satu fitur pada Portal Akademik Mahasiswa namun sekarang fitur tersebut sudah tidak ada lagi. Mahasiswa menginginkan fitur kalender akademik kembali untuk mengetahui tanggal-tanggal penting pada perkuliahan.

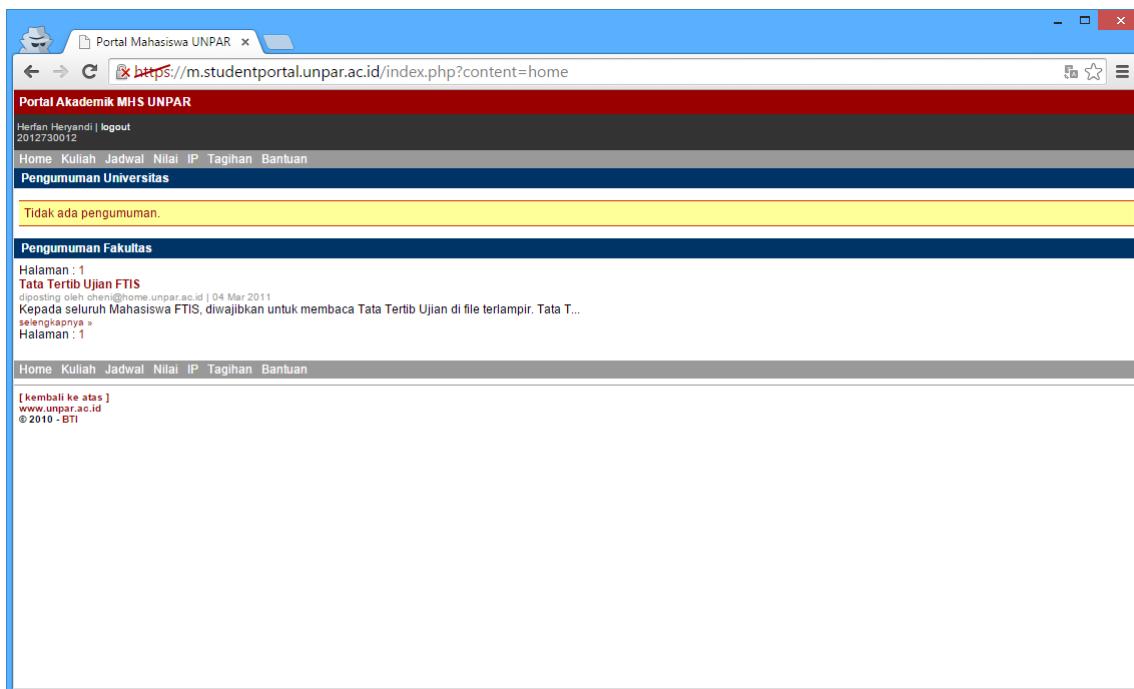
6. Rincian pembayaran

Tagihan pada Portal Akademik Mahasiswa tidak mencantumkan batas akhir pembayaran dan rincian nominal tagihan. Mahasiswa menginginkan rincian pembayaran agar tidak terlambat membayar uang kuliah dan dapat mengetahui rincian nominal tagihan.

7. Rincian mata kuliah

Setiap mata kuliah yang dibuka memiliki rincian seperti deskripsi mata kuliah dan jenis mata kuliah yaitu apakah mata kuliah tersebut wajib, pilihan, atau pilihan wajib. Mahasiswa menginginkan fitur ini agar dapat mengetahui mata kuliah apa yang akan dipelajari.

- 1 8. Tampilan situs web sama di sistem operasi manapun
 2 Jika tidak menggunakan sistem operasi Windows seperti Linux dan Mac, saat meng-
 3 akses Portal Akademik Mahasiswa melalui <https://studentportal.unpar.ac.id/>,
 4 maka mahasiswa akan diarahkan ke <https://m.studentportal.unpar.ac.id/> yaitu
 5 Portal Akademik Mahasiswa dengan tampilan *mobile* (Gambar 3.26). Tampilan ini
 6 tidak memiliki fitur selengkap Portal Akademik Mahasiswa, hanya memiliki fitur pe-
 7 ngumuman kuliah, jadwal kuliah, UTS, dan UAS, nilai, IP, dan tagihan. Selain itu,
 8 tampilan *mobile* pada telepon seluler akan terlihat sangat kecil sehingga tidak sulit
 9 untuk memilih menu. Mahasiswa menginginkan fitur ini agar IT Student Portal dapat
 10 diakses di sistem operasi manapun tanpa perubahan tampilan.



Gambar 3.26: Tampilan *Mobile* Portal Akademik Mahasiswa

- 11 9. Kontak dosen
 12 Kontak dosen berisi informasi email setiap dosen sehingga dapat mempermudah ma-
 13 hasiswa untuk menghubungi dosen. Mahasiswa juga dapat mengirim email secara
 14 langsung melalui Portal Akademik Mahasiswa.
 15 10. Pohon kurikulum
 16 Mahasiswa mungkin ingin agar dapat melihat pohon kurikulum Program Studi Tek-
 17 nik Informatika dalam Portal Akademik Mahasiswa.
 18 11. Pemberitahuan
 19 Mahasiswa menginginkan Portal Akademik Mahasiswa menampilkan pemberitahuan
 20 berupa *pop up* mengenai pengumuman terkini.
 21 12. *Chatting*
 22 Mahasiswa menginginkan agar dapat berkomunikasi dengan sesama pengguna Portal
 23 Akademik Mahasiswa melalui *chat*.

13. Unggah *Curriculum Vitae* Mahasiswa menginginkan agar dapat mengunggah data
 2 mengenai kegiatan dan keaktifan di universitas agar dapat digunakan oleh perusa-
 3 haan untuk mencari mahasiswa dengan kriteria tertentu misalnya untuk kepentingan
 4 magang dan beasiswa.

5 Fitur-fitur yang akan dipilih untuk diimplementasikan harus memenuhi kriteria:

- 6 • Data yang dibutuhkan dapat diambil dari Portal Akademik Mahasiswa
- 7 • Fitur tidak tersedia di Portal Akademik Mahasiswa
- 8 • Fitur mendukung fungsi Portal Akademik Mahasiswa sebagai sumber informasi aka-
 9 demik

10 Hasil analisis fitur-fitur yang diinginkan berdasarkan kriteria di atas dan batas waktu
 11 pembangunan aplikasi dapat dilihat pada tabel 3.1.

Fitur	Dibuat/Tidak dibuat	Alasan
Prasyarat mata kuliah	Dibuat	Prasyarat mata kuliah Program Studi Teknik Informatika sudah tersedia di SIA Models
Ringkasan data akademik	Dibuat	Data dapat diambil dari Portal Akademik Mahasiswa dan didukung oleh SIA Models
Perubahan IPS dan IPK berdasarkan riwayat nilai	Dibuat	IPS dan IPK dapat dihitung melalui riwayat nilai yang dapat diperoleh dari Portal Akademik Mahasiswa
Jadwal kuliah yang tersusun	Dibuat	Jadwal yang tersusun mempermudah mahasiswa untuk
Kalender akademik	Tidak dibuat	Data tidak bisa diperoleh dari Portal Akademik Mahasiswa
Rincian pembayaran	Tidak dibuat	Data tidak bisa diperoleh dari Portal Akademik Mahasiswa
Rincian mata kuliah	Tidak dibuat	Data tidak bisa diperoleh dari Portal Akademik Mahasiswa
Tampilan situs web sama di sistem operasi manapun	Dibuat	Aplikasi yang akan dibuat merupakan situs web yang responsif
Kontak dosen	Tidak dibuat	Data tidak bisa diperoleh dari Portal Akademik Mahasiswa
Pemberitahuan	Tidak dibuat	Waktu pengerjaan yang terbatas
<i>Chatting</i>	Tidak dibuat	Waktu pengerjaan yang terbatas
Unggah <i>Curriculum Vitae</i>	Tidak dibuat	Tidak mendukung Portal Akademik Mahasiswa sebagai sumber informasi akademik

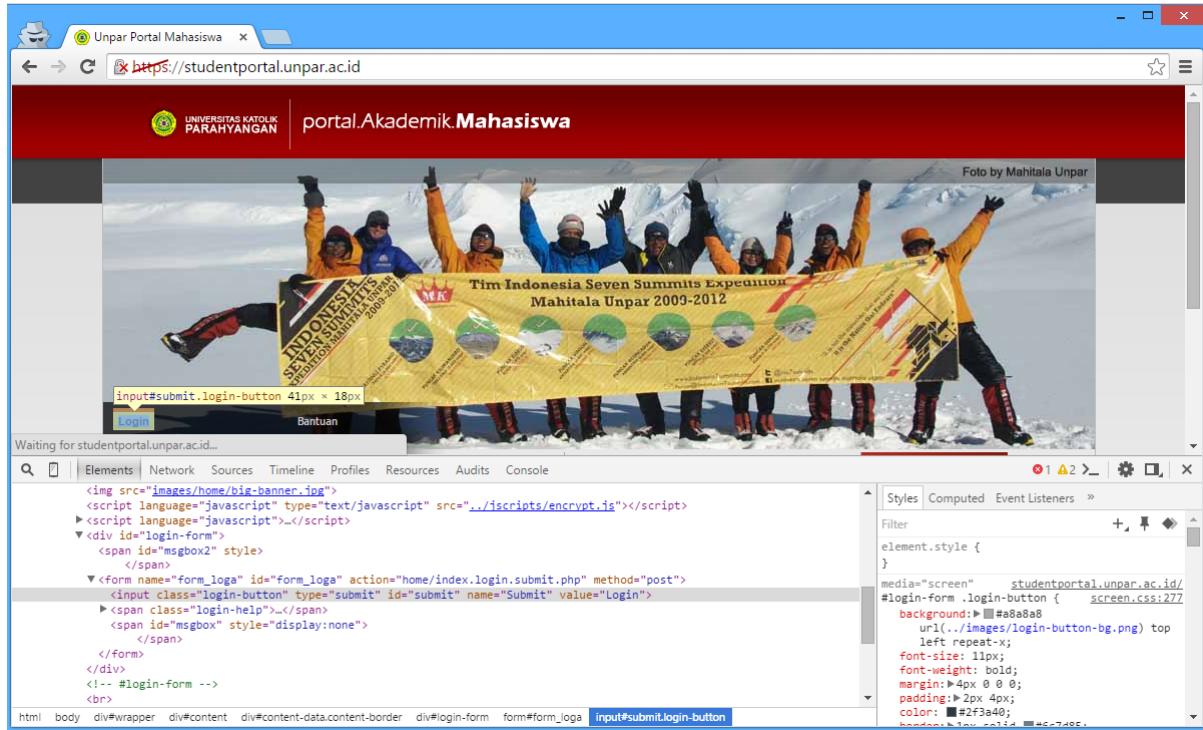
Tabel 3.1: Tabel Hasil Analisis Kebutuhan IT Student Portal

12 3.3 Analisis Komunikasi Portal Akademik Mahasiswa untuk 13 Fitur IT Student Portal

14 Untuk memenuhi fitur IT Student Portal, penulis menganalisis komunikasi Portal Akademik
 15 Mahasiswa ke dalam beberapa kasus yang akan dijelaskan pada subbab-subbab berikut.

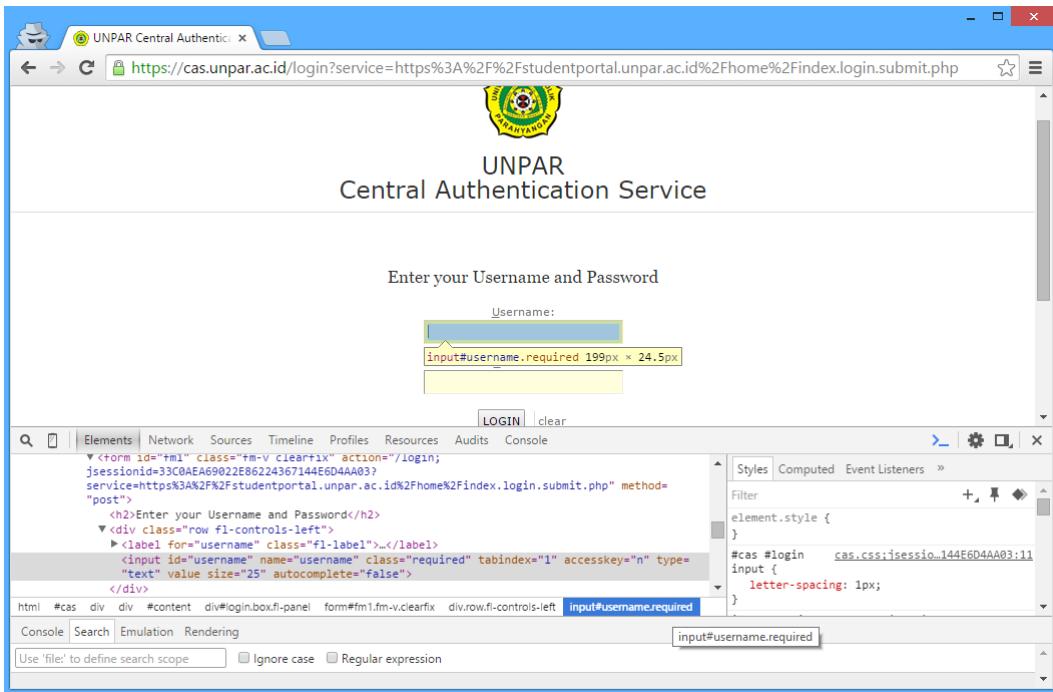
1 3.3.1 Kasus Login

- 2 Di Portal Akademik Mahasiswa, mahasiswa dapat *login* dengan cara:
- 3 1. Mengakses <https://studentportal.unpar.ac.id/> dan mengklik tombol “input#submit.login-button” (Gambar 3.27).



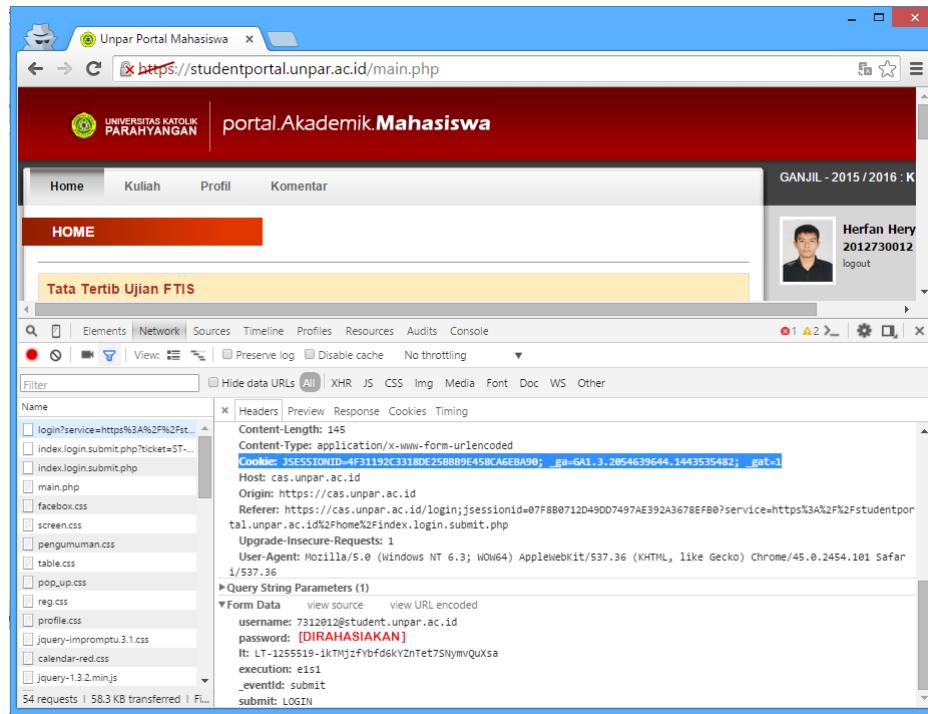
Gambar 3.27: Tombol “input#submit.login-button” pada Halaman Depan Portal Akademik Mahasiswa

- 5 2. Saat tombol tersebut ditekan, mahasiswa akan dibawa ke halaman <index.login.submit.php> dengan *form data* berisi:
- 6
 - 7 • Submit: selalu berisi “Login”
- 8 3. Secara otomatis halaman akan berpindah lagi ke <https://cas.unpar.ac.id/login?service=https%3A%2F%2Fstudentportal.unpar.ac.id%2Fhome%2Findex.login.submit.php>.
- 10 4. Di sana, akan ditampilkan halaman *login* CAS UNPAR di mana mahasiswa diminta mengisi “Username” pada kolom “input#username.required” dan “Password” pada kolom “input#password.required” (Gambar 3.28).



Gambar 3.28: Kolom “Username” “input#username.required” pada Halaman CAS UNPAR

- 1 5. Setelah itu mahasiswa harus menekan tombol “input.btn-submit”. Data tersebut akan
- 2 dikirimkan ke [/login;jsessionid=...?service=https://studentportal.unpar.ac.id/home/index.login.submit.php](https://cas.unpar.ac.id/login?service=https%3A%2F%2Fstudentportal.unpar.ac.id%2Fhome%2Findex.login.submit.php) dengan *cookie*:
- 3
 - 4 • JSESSIONID: diambil dari *cookie* yang di-set pada halaman <https://cas.unpar.ac.id/login?service=https%3A%2F%2Fstudentportal.unpar.ac.id%2Fhome%2Findex.login.submit.php>
- 7 Data yang dikirim juga mengandung *form data* sebagai berikut (Gambar 3.29):
 - 8 • username: diambil dari nilai elemen “input#username.required”
 - 9 • password: diambil dari nilai elemen “input#password.required”
 - 10 • lt: diambil dari nilai elemen “input” dengan nama “lt”
 - 11 • execution: diambil dari nilai elemen “input” dengan nama “execution”
 - 12 • _eventId: selalu berisi “submit”
 - 13 • submit: selalu berisi “LOGIN”



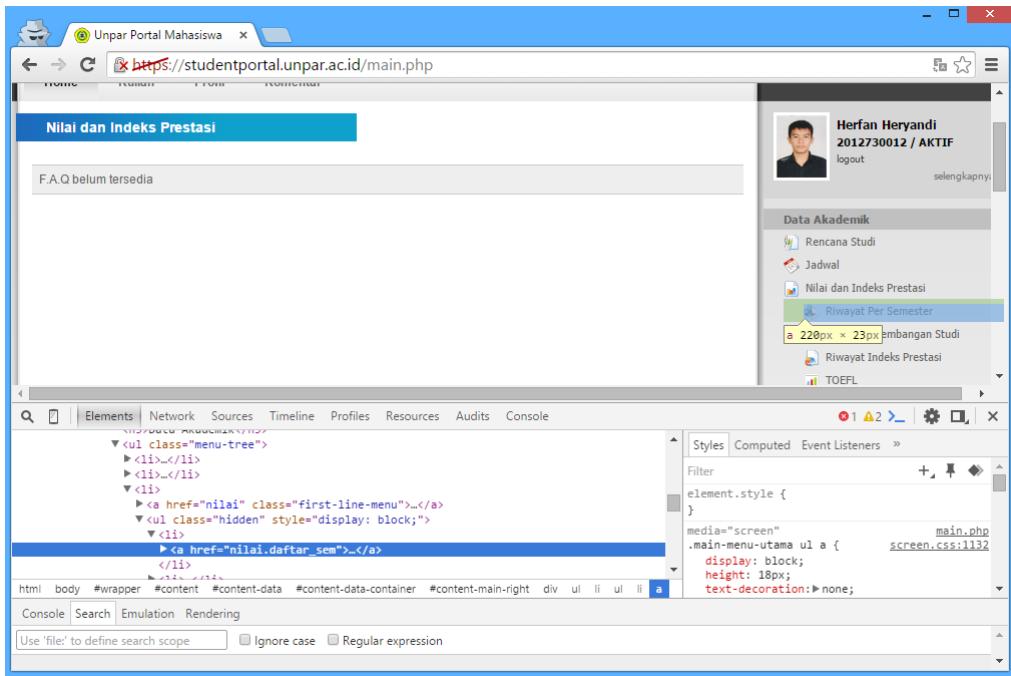
Gambar 3.29: *Form Data* yang dikirim CAS UNPAR

- 1 6. Jika berhasil, akan dilakukan pengalihan beberapa kali dan diakhiri di <https://studentportal.unpar.ac.id/main.php> dengan *cookie* sebagai berikut:
- 2
 - 3 • PHPSESSID: diambil dari *cookie* yang di-set pada beberapa pengalihan sebelumnya
 - 4 • _ga dan _gat: tidak diperlukan, digunakan oleh Google Analytics²

6 3.3.2 Kasus Nilai

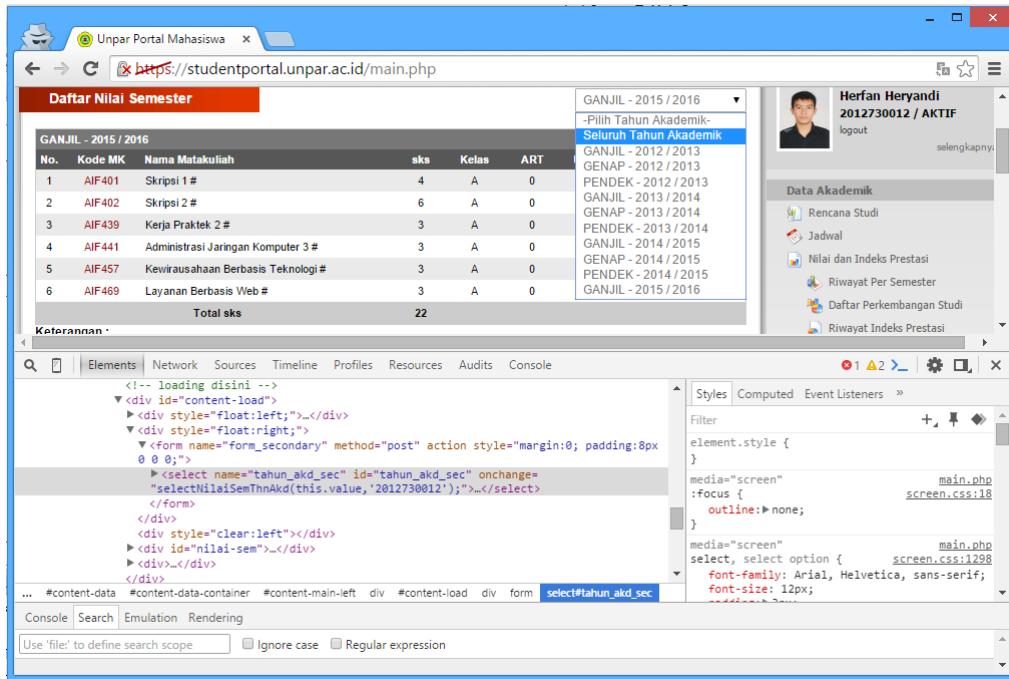
- 7 Di halaman utama Portal Akademik Mahasiswa, mahasiswa dapat melihat riwayat nilai dengan cara:
 - 9 1. Mengklik “a.first-line-menu” dengan teks “Nilai dan Indeks Prestasi”
 - 10 2. Setelah diklik, akan muncul list “ul.hidden”, kemudian mahasiswa harus mengklik elemen “a” dengan teks “Riwayat Per Semester” (Gambar 3.30)

²<https://developers.google.com/analytics/devguides/collection/analyticsjs/cookie-usage>



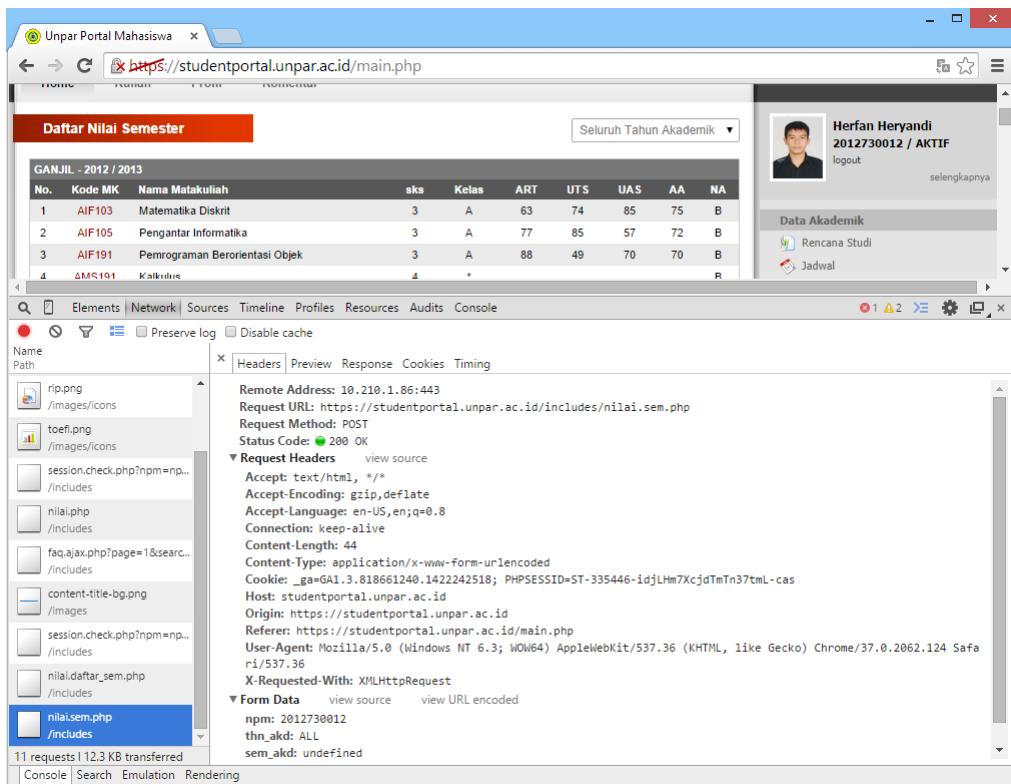
Gambar 3.30: Elemen “a” dengan teks “Riwayat Per Semester” pada Menu Nilai dan Indeks Prestasi

- 1 3. Setelah mengklik “Riwayat Per Semester”, mahasiswa akan diarahkan ke https://studentportal.unpar.ac.id/includes/nilai.daftar_sem.php dengan *cookie* sebagai berikut:
- 2 • PHPSESSID: diambil dari *cookie* yang di-set saat pengalihan ke halaman utama
- 3 4. Halaman “Riwayat Per Semester” menampilkan nilai semester terkini. Jika ingin melihat nilai semester sebelumnya, mahasiswa dapat mengklik *combo box* “select#tahun_akd_sec” kemudian memilih “option” yang diinginkan atau “Seluruh Tahun Akademik” untuk melihat nilai seluruh semester (Gambar 3.31).



Gambar 3.31: *Combo Box “select#tahun_akd_sec” pada Halaman Riwayat Per Semester*

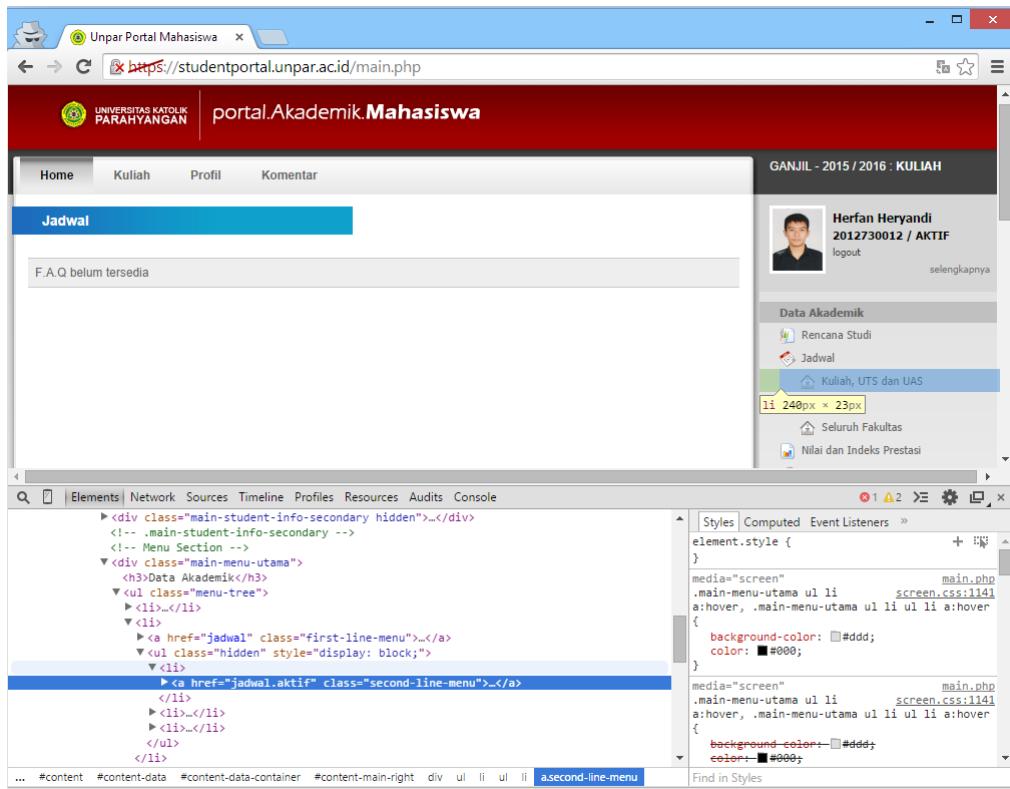
- 1 5. Setelah memilih “option”, mahasiswa akan dibawa ke <https://studentportal.unpar.ac.id/includes/nilai.sem.php> (Gambar 3.32) dengan cookie PHPSESSID dan mengandung *form data*:
- 2 6
 - npm: diperoleh dari NPM mahasiswa
 - thn_akd: berisi tahun akademik semester atau berisi “ALL” jika memilih “Seluruh Tahun Akademik”
 - sem_akd: berisi semester akademik dalam angka (1: ganjil, 2: genap, 4: pendek) atau tidak didefinisikan jika memilih “Seluruh Tahun Akademik”



Gambar 3.32: *Form Data* pada pengiriman Nilai Seluruh Tahun Akademik

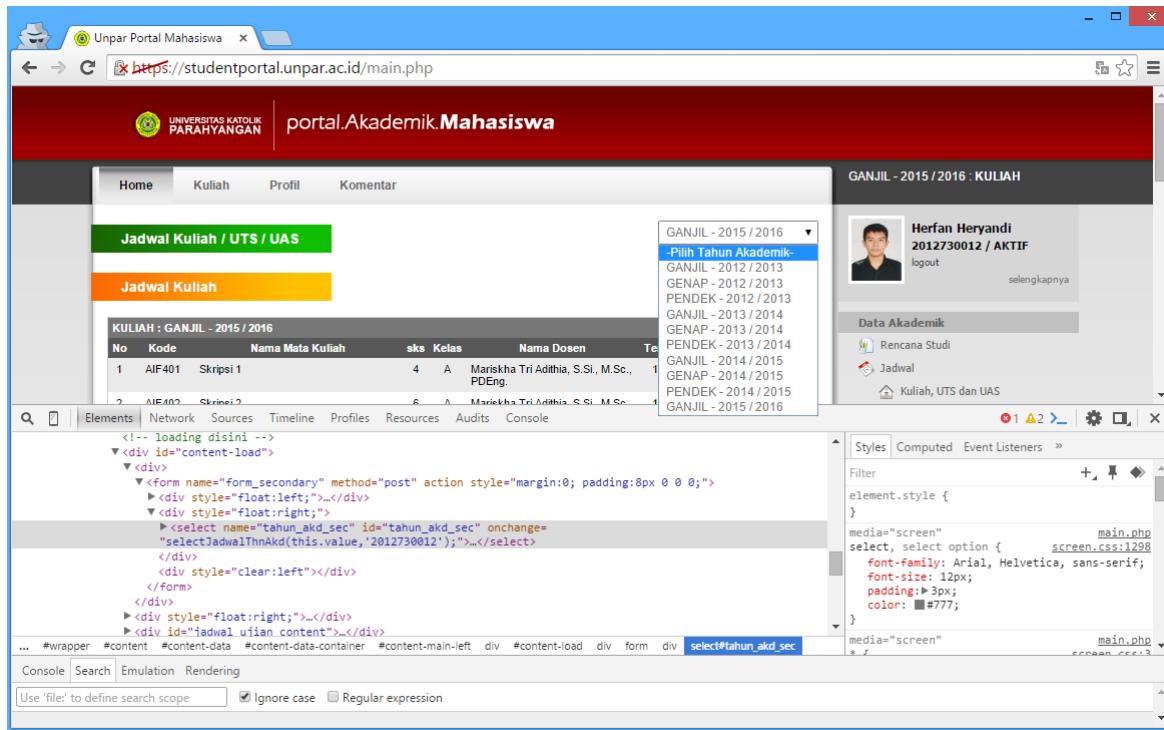
3.3.3 Kasus Jadwal

- 2 Di halaman utama Portal Akademik Mahasiswa, mahasiswa dapat melihat jadwal dengan cara:
- 4 1. Mengklik “a.first-line-menu” dengan teks “Jadwal”
- 5 2. Setelah diklik, akan muncul list “ul.hidden”, kemudian mahasiswa harus mengklik elemen “a” dengan teks “Kuliah, UTS dan UAS”(Gambar 3.33)



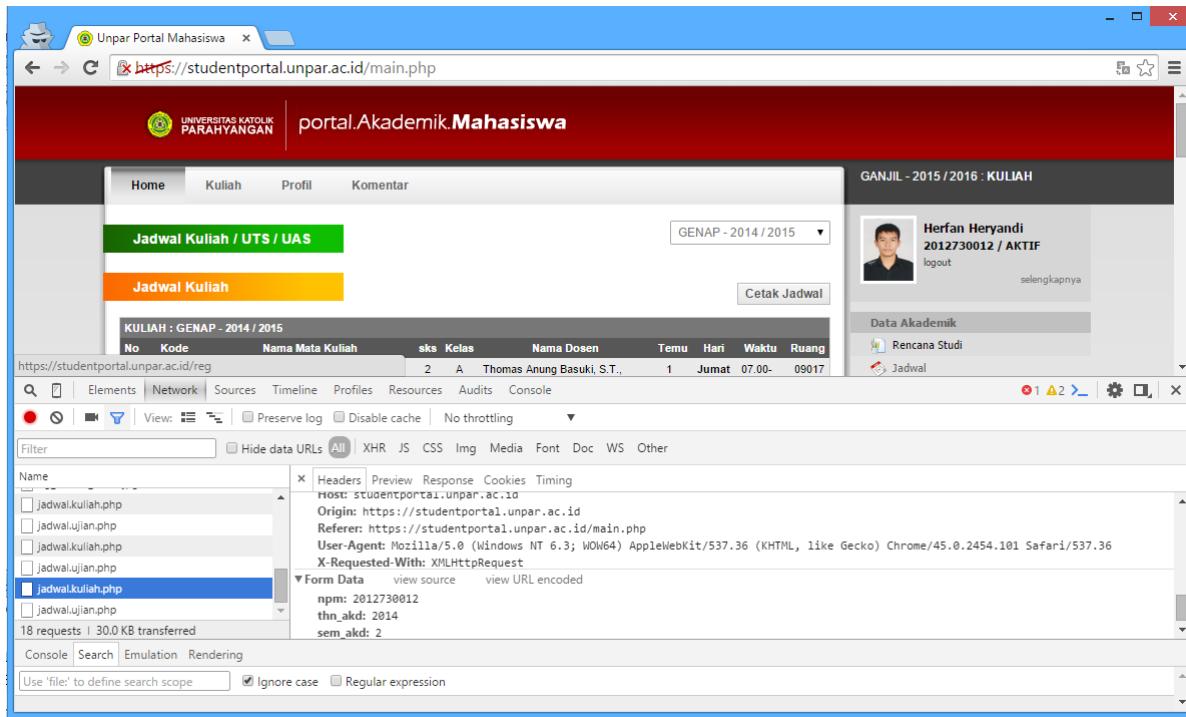
Gambar 3.33: Elemen “a” dengan teks “Kuliah, UTS dan UAS” pada Menu Jadwal

- 1 3. Setelah mengklik “Kuliah, UTS dan UAS”, mahasiswa akan diarahkan ke <https://studentportal.unpar.ac.id/includes/jadwal.aktif.php> dengan *cookie* sebagai berikut:
- 2 4. Halaman “Kuliah, UTS dan UAS” menampilkan jadwal kuliah, UTS, dan UAS semester terkini. Jika ingin melihat jadwal semester sebelumnya, mahasiswa dapat mengklik *combo box* “select#tahun_akd_sec” kemudian memilih “option” yang diinginkan (Gambar 3.34).



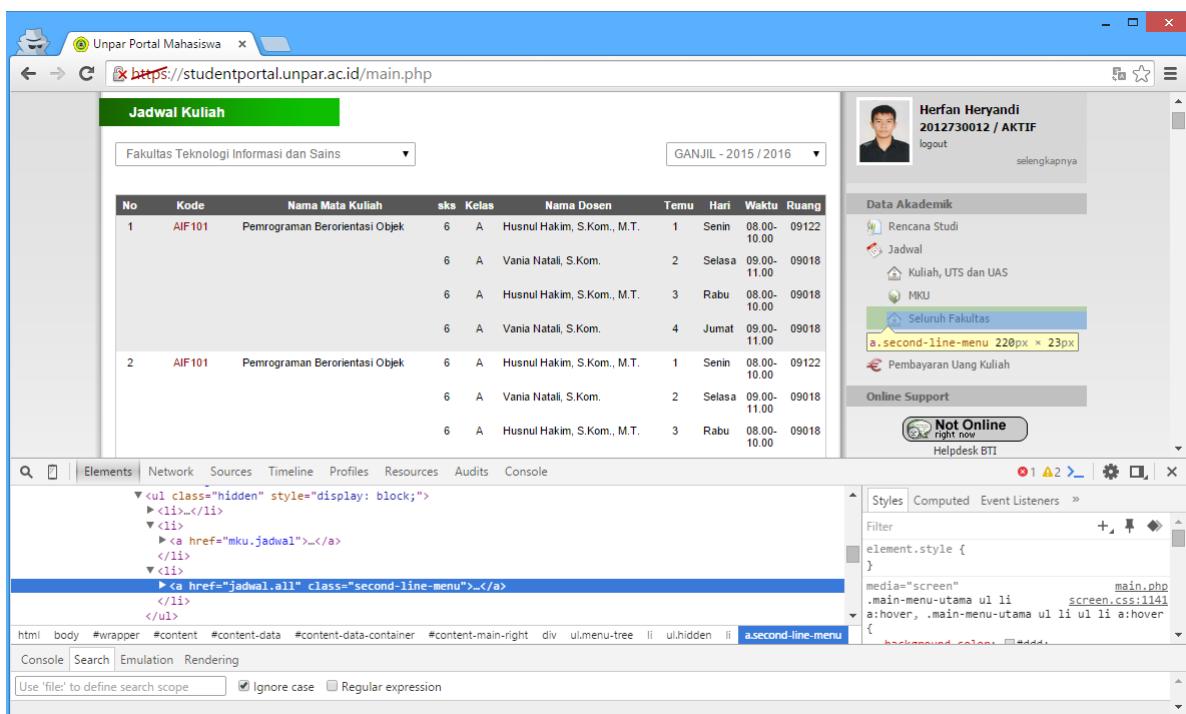
Gambar 3.34: Combo Box “select#tahun_akd_sec” pada Halaman Jadwal Kuliah, UTS, dan UAS

- 1 5. Setelah memilih “option”, mahasiswa akan ditampilkan jawaban dari <https://studentportal.unpar.ac.id/includes/jadwal.kuliah.php> dan <https://studentportal.unpar.ac.id/includes/jadwal.ujian.php> (Gambar 3.35) dengan cookie PHPSESSID dan mengandung *form data*:
- 2 6.
 - 5 • npm: diperoleh dari NPM mahasiswa
 - 6 • thn_akd: berisi tahun akademik semester
 - 7 • sem_akd: berisi semester akademik dalam angka (1: ganjil, 2: genap, 4: pendek)



Gambar 3.35: *Form Data* pada pengiriman Jadwal Kuliah dan Ujian

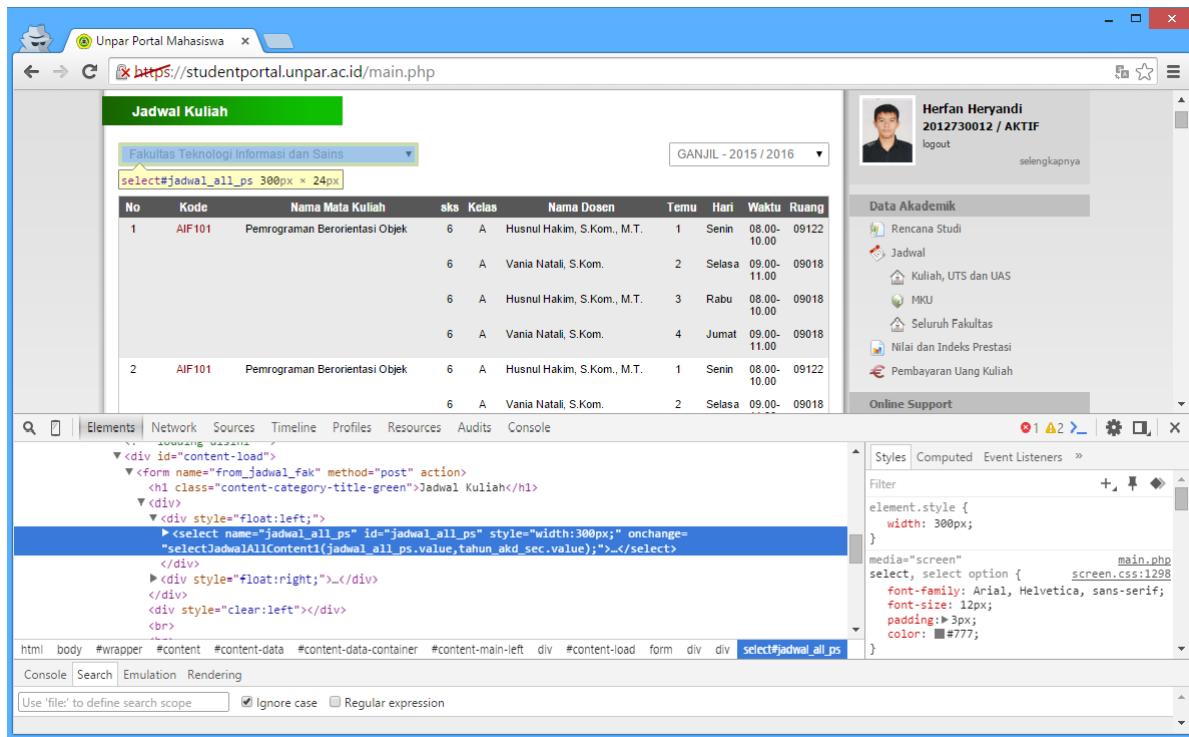
- 1 Mahasiswa juga dapat melihat jadwal seluruh fakultas dengan cara:
- 2 Mengklik “a.second-line-menu” dengan teks “Seluruh Fakultas” pada *list “ul.hidden”*
- 3 (Gambar 3.36)



Gambar 3.36: Elemen “a” dengan teks “Seluruh Fakultas” pada Menu Jadwal

- 4 Setelah diklik, mahasiswa akan diarahkan ke <https://studentportal.unpar.ac.id/includes/jadwal.all.php> dengan *cookie* sebagai berikut:
- 5

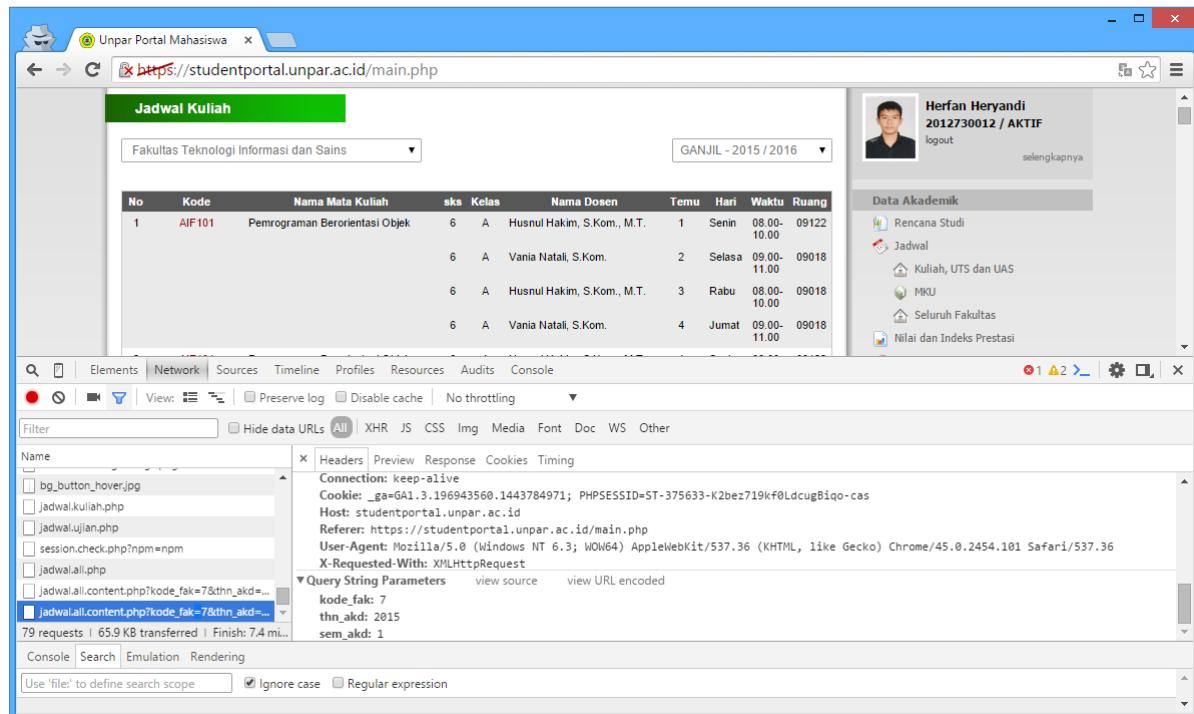
- PHPSESSID: diambil dari *cookie* yang di-set saat pengalihan ke halaman utama
3. Halaman “Seluruh Fakultas” menampilkan seluruh jadwal semester terkini pada fakultas tempat mahasiswa menempuh studi. Jika ingin melihat jadwal semester sebelumnya, mahasiswa dapat mengklik *combo box* “select#tahun_akd_sec” kemudian memilih “option” yang diinginkan. Jika ingin melihat jadwal fakultas lain, mahasiswa dapat mengklik *combo box* “select#jadwal_all_ps” kemudian memilih “option” fakultas yang diinginkan (Gambar 3.37).



Gambar 3.37: Combo Box “select#jadwal_all_ps” pada Halaman Jadwal Seluruh Fakultas

- 8 4. Setelah memilih “option”, mahasiswa akan dibawa ke <https://studentportal.unpar.ac.id/includes/jadwal.all.content.php>(Gambar 3.38) dengan *cookie* PHPSESSID dan mengandung *query string parameter*:

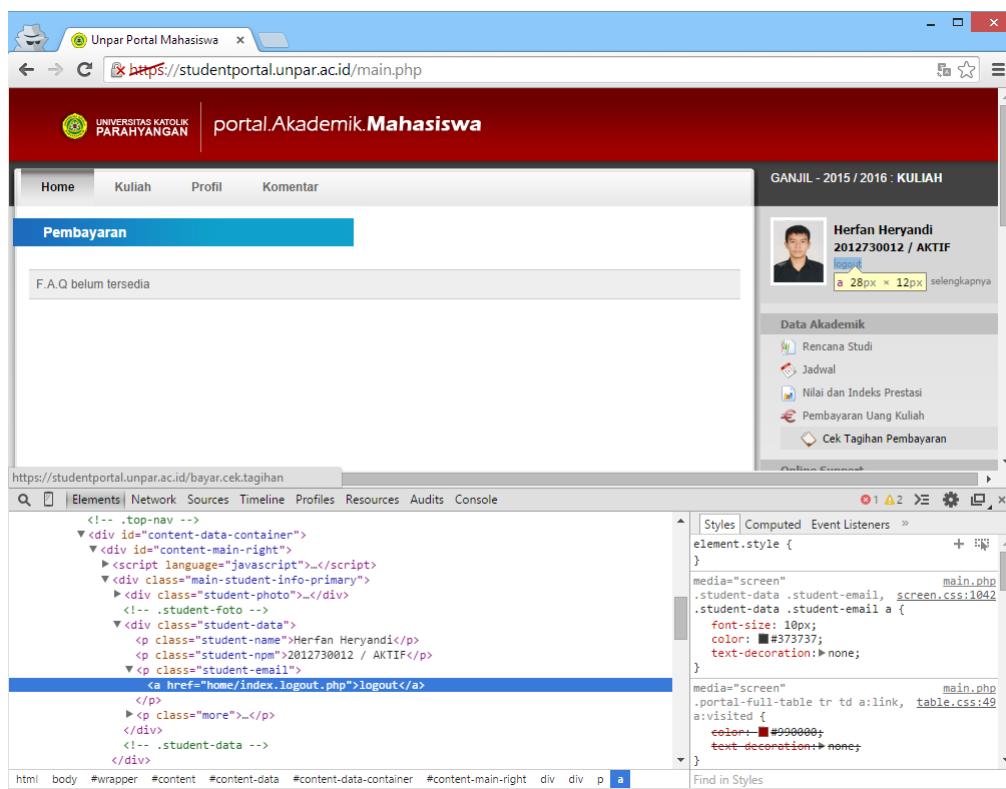
- kode_fak: berisi kode fakultas
- thn_akd: berisi tahun akademik semester
- sem_akd: berisi semester akademik dalam angka (1: ganjil, 2: genap, 4: pendek)



Gambar 3.38: Form Data pada pengiriman Jadwal Seluruh Fakultas

3.3.4 Kasus Logout

- 2 Mahasiswa dapat melakukan *logout* dengan cara:
- 3 1. Mengklik “a” dengan teks “logout” pada bagian identitas portal (Gambar 3.39)

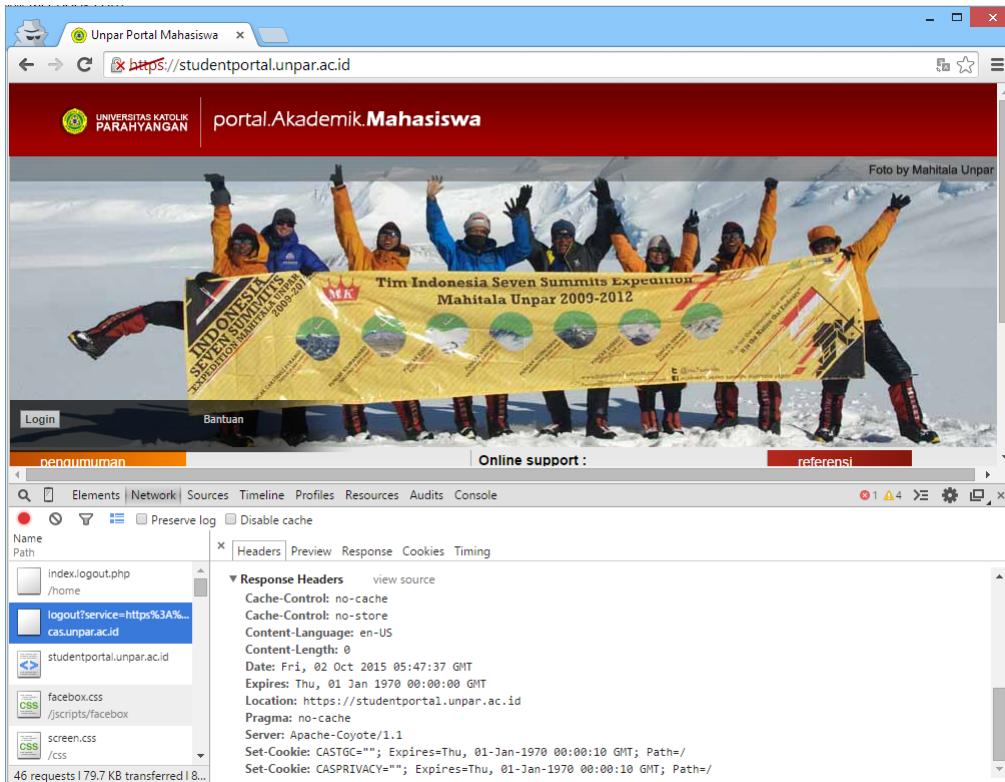


Gambar 3.39:

- 4 2. Setelah diklik, mahasiswa akan diarahkan ke <https://studentportal.unpar.ac.id/>

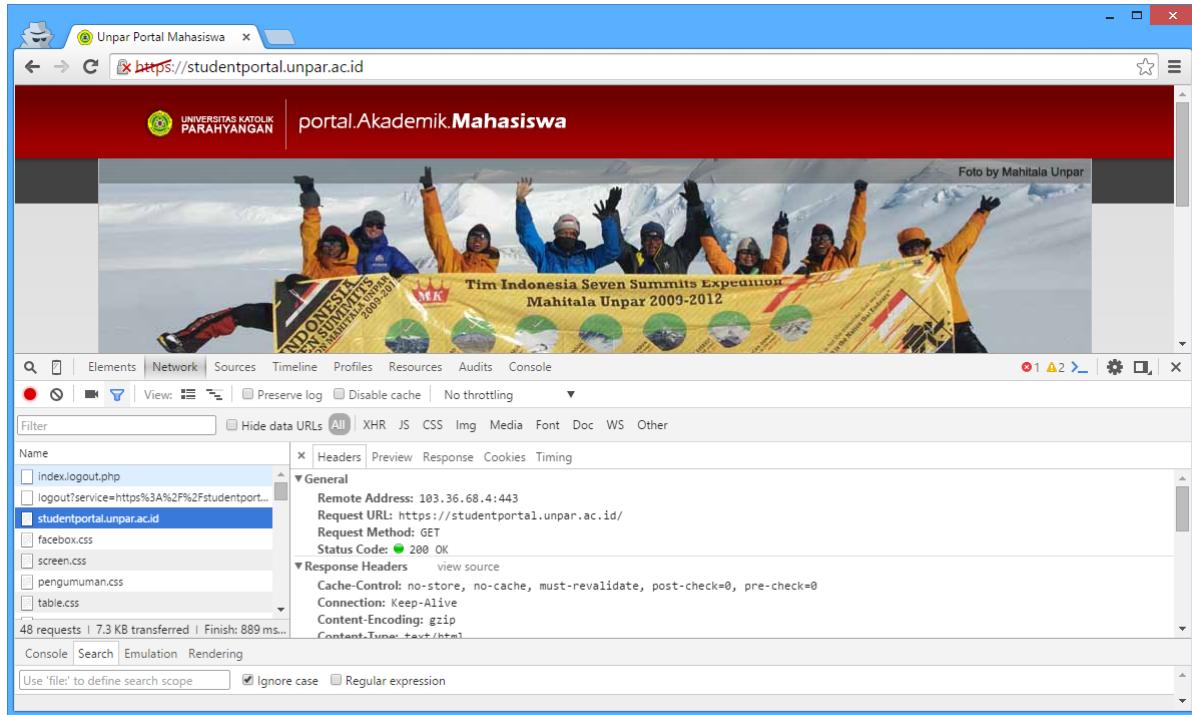
1 <home/index.logout.php> dengan *cookie* sebagai berikut:

- 2 • PHPSESSID: diambil dari *cookie* yang di-set saat pengalihan ke halaman utama
- 3 3. Kemudian dilakukan pengalihan ke <https://cas.unpar.ac.id/logout?service=https%3A%2F%2Fstudentportal.unpar.ac.id> lalu mengadaluarsakan *cookie* CASTGC dan CASPRIVACY yang sebelumnya di-set saat pengalihan ke halaman utama (Gambar 3.40)
- 4
- 5
- 6



Gambar 3.40: Pengadaluarsaan Cookie CASTGC dan CASPRIVACY

- 7 4. Pengalihan diakhiri di <https://studentportal.unpar.ac.id/> yaitu halaman depan
- 8 Portal Akademik Mahasiswa (Gambar 3.41)

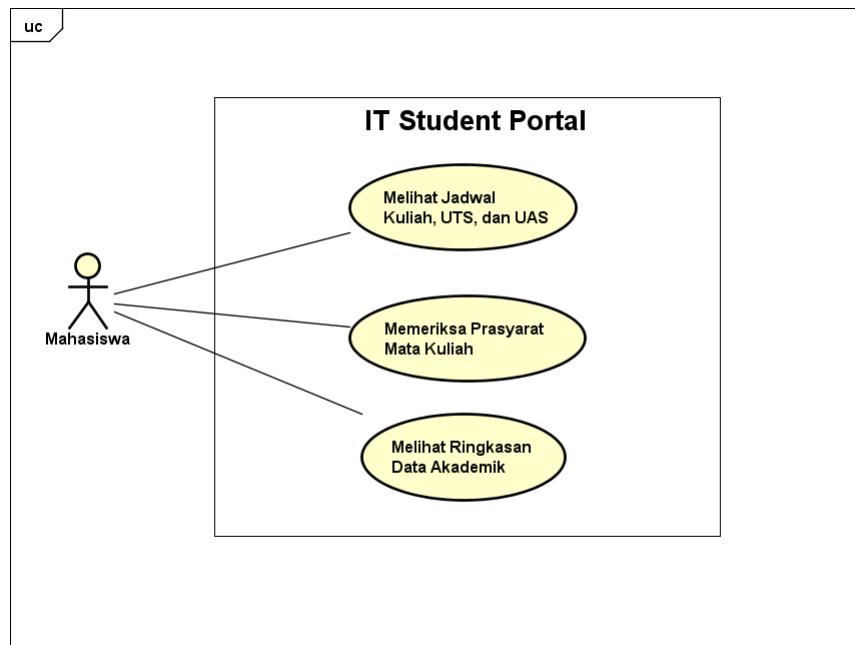


Gambar 3.41: Pengalihan ke Halaman Depan Portal Akademik Mahasiswa

¹ 3.4 Analisis Use Case

² 3.4.1 Diagram Use Case

- ³ Diagram *use case* pada perangkat lunak yang akan dibangun hanya mengandung satu aktor, yaitu mahasiswa. Diagram *use case* dapat dilihat pada gambar 3.42.



Gambar 3.42: Diagram Use Case IT Student Portal

- ⁵ Berdasarkan hasil analisis pada subbab 3.2, dari lima fitur yang akan dibuat, dibentuk

1 tiga *use case* antara lain:

- 2 • **Memeriksa Prasyarat Mata Kuliah**, mahasiswa dapat memeriksa mata kuliah
3 yang dibuka pada semester terkini apakah memenuhi prasyarat atau tidak.
- 4 • **Melihat Jadwal Kuliah, UTS, dan UAS**, mahasiswa dapat melihat jadwal kuliah,
5 UTS, dan UAS yang sudah tersusun dan terurut berdasarkan hari.
- 6 • **Melihat Ringkasan Data Akademik**, mahasiswa dapat melihat data mengenai ma-
7 taa kuliah apa saja yang sudah lulus beserta jenis mata kuliahnya(wajib, pilihan, atau
8 pilihan wajib), sisa SKS untuk mencapai kelulusan, dan mata kuliah wajib yang be-
9 lum ditempuh. Mahasiswa juga dapat melihat IPS dan IPK yang berubah berdasarkan
10 riwayat nilai.

11 3.4.2 Skenario Use Case

12 1. Memeriksa Prasyarat Mata Kuliah

- 13 • Nama: Memeriksa prasyarat mata kuliah
- 14 • Aktor: Mahasiswa
- 15 • Deskripsi: Memeriksa prasyarat mata kuliah yang dibuka pada semester terkini
- 16 • Kondisi awal: Mahasiswa telah *login*
- 17 • Kondisi akhir: Halaman prasyarat mata kuliah ditampilkan dan berisi prasyarat
18 mata kuliah yang dibuka pada semester terkini
- 19 • Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu prasyarat mata kuliah.	Sistem mendapatkan data mahasiswa kemudian menampilkan halaman prasyarat mata kuliah

- 20 • Eksepsi: -

21 2. Melihat Jadwal Kuliah, UTS, dan UAS

- 22 • Nama: Melihat jadwal kuliah, UTS, dan UAS
- 23 • Aktor: Mahasiswa
- 24 • Deskripsi: Melihat jadwal kuliah, UTS, dan UAS yang sudah tersusun dan terurut
25 berdasarkan hari
- 26 • Kondisi awal: Mahasiswa telah *login*
- 27 • Kondisi akhir: Halaman jadwal ditampilkan dan berisi jadwal kuliah, UTS, dan
28 UAS yang sudah tersusun dan terurut berdasarkan hari te
- 29 • Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu jadwal.	Sistem menyusun dan mengurutkan jadwal mahasiswa berdasarkan hari kemudian menampilkan halaman ja-dwal

- 1 • Eksepsi: Mahasiswa sedang cuti studi

2 **3. Melihat Ringkasan Data Akademik**

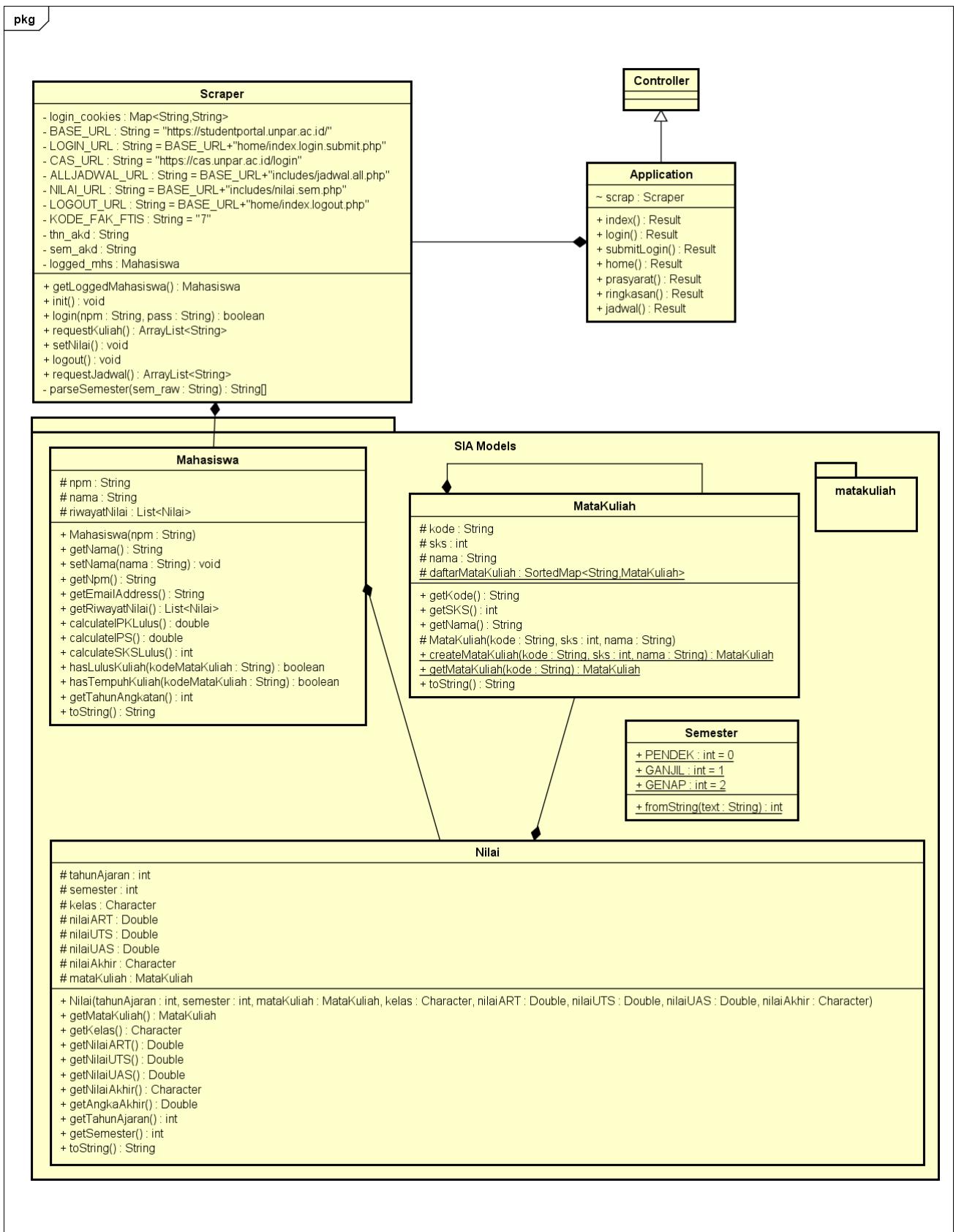
- 3 • Nama: Melihat ringkasan data akademik
- 4 • Aktor: Mahasiswa
- 5 • Deskripsi: melihat data mengenai mata kuliah apa saja yang sudah lulus beserta jenis mata kuliahnya(wajib, pilihan, atau pilihan wajib), sisa SKS untuk mencapai kelulusan, dan mata kuliah wajib yang belum ditempuh. Mahasiswa juga dapat melihat IPS dan IPK yang berubah berdasarkan riwayat nilai
- 6 • Kondisi awal: Mahasiswa telah *login*
- 7 • Kondisi akhir: Halaman jadwal ditampilkan dan berisi jadwal ringkasan data akademik
- 8 • Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu ringkasan data akademik.	Sistem meringkas data kademik mahasiswa kemudian menampilkan halaman ringkasan data akademik

- 13 • Eksepsi: -

14 **3.5 Analisis Kelas**

15 Diagram kelas analisis untuk IT Student Portal ditunjukkan pada gambar [3.43](#).



Gambar 3.43: Diagram Kelas Analisis IT Student Portal

- 1 Kelas-kelas dari package SIA Models telah dibahas pada subbab 2.4. Sedangkan enjelasan
- 2 dari kelas-kelas lainnya sebagai berikut:

1. Controller
2. Kelas ini merupakan kelas yang sudah disediakan oleh Play Framework. Turunan dari kelas ini akan berperan sebagai *controller*.
4. Application
5. Kelas ini merupakan controller pada IT Student Portal yang berfungsi untuk menghubungkan *model* dengan *view*.
7. Scraper
8. Kelas ini merupakan kelas yang mengimplementasikan *web scraping* menggunakan *library* jsoup. Kelas ini digunakan untuk memperoleh data dari Portal Akademik Mahasiswa kemudian menghubungkannya ke dalam SIA Models. Data-data yang sudah dihubungkan dengan SIA Models akan ditampilkan ke *view* melalui kelas Application.

1

DAFTAR REFERENSI

- 2 [1] Program Studi Teknik Informatika, “Prasyarat Mata Kuliah Semester Ganjil 2015/2016.”
3 <https://tinyurl.com/lionov>, Juli 2015. [Online; diakses 1-Juli-2015].
- 4 [2] Biro Teknologi Informasi UNPAR, “Portal Akademik Mahasiswa.” <https://studentportal.unpar.ac.id>, 2012. [Online; diakses 1-Juli-2015].
- 6 [3] N. Leroux and S. de Kaper, *Play for Java*. Manning Publications Co., 2014.
- 7 [4] Biro Teknologi Informasi UNPAR, *Buku Panduan Layanan TIK 2012*. Biro Teknologi
8 Informasi UNPAR, 2012.
- 9 [5] J. Hedley, “jsoup: Java HTML Parser.” <http://jsoup.org>, 2009-2015. [Online; diakses
10 1-Juli-2015].
- 11 [6] P. Alfadian, “SIA Models.” <https://github.com/pascalalfadian/SIAModels>, 2015.
12 [Online; diakses 1-Juli-2015].
- 13 [7] E. Vargiu and M. Urru, “Exploiting web scraping in a collaborative filtering-based approach to web advertising,” in *Artificial Intelligence Research*, (Barcelona,Spain), pp. 44–50,
14 2013.
- 16 [8] Google, “Chrome DevTools.” <https://developer.chrome.com/devtools>, 2013. [Online;
17 diakses 1-Juli-2015].
- 18 [9] E. A. Meyer, *Selectors, Specificity, and the Cascade*. O'Reilly Media, 2012.