

**ADOPSI METODE *FEATURE-DRIVEN*
DEVELOPMENT DALAM RANCANG
BANGUN SISTEM AUCTENTIK**



Disusun Oleh:

N a m a : Yoga Aryapratama

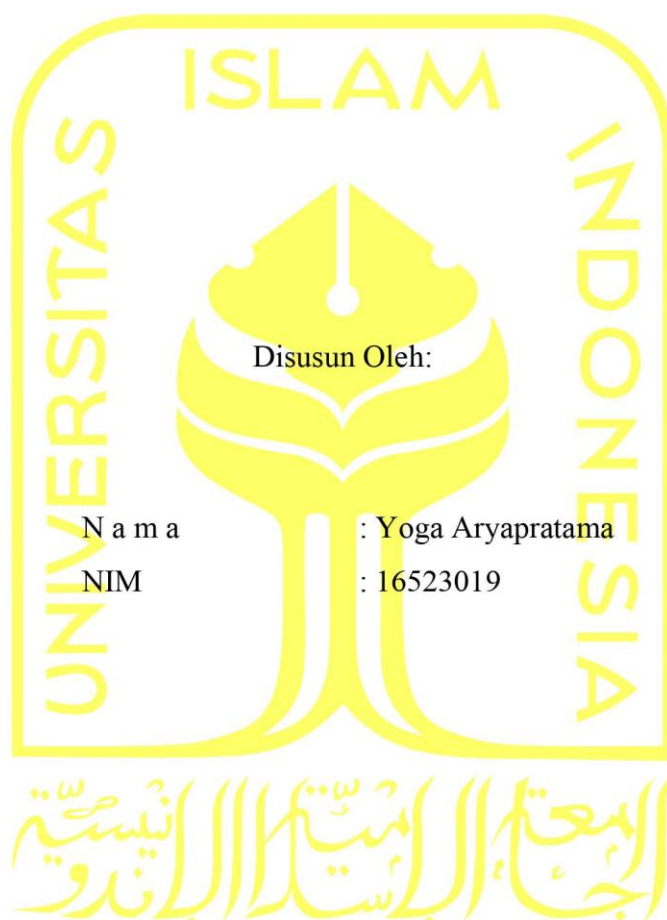
NIM : 16523019

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2020

HALAMAN PENGESAHAN DOSEN PEMBIMBING
ADOPSI METODE *FEATURE-DRIVEN*
***DEVELOPMENT* DALAM RANCANG**
BANGUN SISTEM AUCTENTIK

TUGAS AKHIR



Yogyakarta, 10 Agustus 2020

Pembimbing,

(Beni Suranto, S.T., M.Soft.Eng.)

HALAMAN PENGESAHAN DOSEN PENGUJI

**ADOPTSI METODE *FEATURE-DRIVEN*
DEVELOPMENT DALAM RANCANG
BANGUN SISTEM AUCTENTIK**

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 10 Agustus 2020

Tim Penguji

Beni Suranto, S.T., M.Soft.Eng.

Anggota 1

Aridhanyati Arifin, S.T., M.Cs.

Anggota 2

Erika Ramadhani, S.T., M.Eng.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Yoga Aryapratama

NIM : 16523019

Tugas akhir dengan judul:

ADOPSI METODE *FEATURE-DRIVEN* *DEVELOPMENT* DALAM RANCANG BANGUN SISTEM AUCTENTIK

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 10 Agustus 2020



(Yoga Aryapratama)

HALAMAN PERSEMBAHAN

Alhamdulillah, segala puji syukur bagi Allah SWT atas segala limpahan ridho, hidayah, dan inayah-Nya sehingga penulis dapat menyelesaikan tugas akhir ini dengan baik dan lancar.

Tugas akhir ini penulis persembahkan untuk:

1. Ibu Nuryani Ekaningsih, yang saya cintai, saya sayangi, saya kasihi, dan saya hormati. Terima kasih atas segala bentuk dukungan dan doa yang diberikan, karena tanpa restu dari Ibu tugas akhir ini tidak akan dapat saya selesaikan.
2. Mbah Nasiyo dan Mbah Rasinem yang selalu mendoakan, mendukung, dan menyayangi saya.
3. Bapak Beni Suranto, S.T., M.Soft.Eng. selaku dosen pembimbing, terima kasih untuk semua nasihat, bimbingan, dan ilmu yang luar biasa yang selama ini telah diberikan kepada saya.
4. Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia yang telah mewadahi saya dalam menuntut ilmu.
5. Seluruh teman-teman saya, terima kasih untuk dukungan, doa, dan kepercayaan bahwa saya bisa menyelesaikan tugas akhir.
6. Serta seluruh pihak yang tidak bisa saya sebutkan satu per satu, terimakasih atas segala dukungan.

HALAMAN MOTO

“Bilamana aku bingung, akupun pergi shalat menghadap maha pencipta, sampai dibukakan-Nya kesulitan dan dimudahkan-Nya kesukaran.”

(Ibnu Sina)

“Simple can be harder than complex: You have to work hard to get your thinking clean to make it simple. But it’s worth it in the end because once you get there, you can move mountains.”

(Steve Jobs)

“Change is definitely every day. It's an ever-changing world. Everything changes, so I really don't know. I just hope I become a better person, a better man.”

(Destin Choice Route)

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakaatuh.

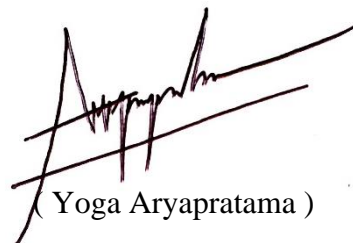
Alhamdulillah, segala puji syukur bagi Allah SWT atas segala limpahan ridho, hidayah, dan inayah-Nya. Sehingga saya dapat menyelesaikan laporan tugas akhir ini yang berjudul “Adopsi Metode *Feature-driven Development* dalam Rancang Bangun Sistem Auctentik”. Laporan ini disusun untuk memenuhi tugas akhir sebagai syarat untuk menyelesaikan pendidikan pada jenjang Strata 1 (S1) pada Jurusan Informatika Fakultas Teknologi Industri Universitas Islam Indonesia. Saya sadar laporan ini tidak akan selesai tepat waktu tanpa dukungan serta motivasi dari beberapa pihak. Oleh karena itu saya tidak lupa menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Allah SWT, atas limpahan rahmat dan hidayah-Nya yang selalu ada di setiap langkah dalam memberikan kekuatan, kemampuan, dan menjaga semangat saya untuk dapat menyelesaikan tugas akhir ini.
2. Ibu Nuryani Ekaningsih untuk doa, dukungan, semangat, dan kepercayaan yang telah diberikan kepada saya selama ini.
3. Bapak Fathul Wahid, S.T., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia.
4. Bapak Prof. Dr. Ir. Hari Purnomo, M.T., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Bapak Hendrik, S.T., M.Eng., selaku Ketua Jurusan Informatika Universitas Islam Indonesia.
6. Bapak Dr. Raden Teduh Dirgahayu, S.T., M.Sc., selaku Ketua Program Studi Informatika Program Sarjana Fakultas Teknologi Industri Universitas Islam Indonesia.
7. Bapak Beni Suranto, S.T., M.Soft.Eng., selaku dosen pembimbing tugas akhir yang memberikan peran luar biasa secara langsung dalam penyelesaian tugas akhir ini.
8. Bapak Hari Setiaji, S.Kom., M.Eng., selaku dosen pembimbing penjaluran perintisan bisnis.
9. Ibu Arrie Kurniawardhani, S.Si., M.Kom., Ibu Nur Wijyaning Rahayu, S.Kom., M.Cs., dan Bapak Hanson Prihantoro Putro, S.T., M.T., selaku dosen penuh jasa yang memotivasi dan memberikan ilmu dan pengalaman yang luar biasa kepada saya.

10. Teman-teman Keluarga Spotify, yang selalu percaya dan mendukung saya dalam pengerjaan tugas akhir ini.
11. Rekan Tim Gaya Sejahtera Nusantara, yang menemani perjalanan penjaluran perintisan bisnis.
12. Teman-teman NB, yang selalu mendukung, memotivasi, dan mendoakan saya dalam menyelesaikan tugas akhir ini.
13. Teman-teman Kandji Tedjo, yang selalu memotivasi dan menghibur saya dalam keadaan apapun.
14. Seluruh pihak yang tidak bisa saya sebutkan satu per satu, terima kasih atas semua bentuk dukungannya.

Semoga segala doa, bantuan, dukungan dan bimbingan yang telah diberikan kepada saya mendapat balasan kebaikan dari Allah SWT. Saya memohon maaf sebesar-besarnya apabila selama mengerjakan Tugas Akhir terdapat banyak kesalahan. Saya menyadari akan keterbatasan kemampuan yang saya miliki. Semoga laporan ini dapat bermanfaat bagi semua yang membacanya. Amin.

Yogyakarta, 10 Agustus 2020



(Yoga Aryapratama)

SARI

Start-up di Indonesia berkembang sangat pesat, salah satu sektor penyumbangannya adalah sektor *e-commerce* dan *e-marketplace*. *Fashion* merupakan komoditas yang memiliki penjualan yang tinggi pada sektor tersebut, termasuk diantaranya kategori “*hypebeast*” dan “terbatas”, sehingga sistem lelang menjadi metode yang dipilih untuk dapat menjual barang tersebut. Auctentik dirintis untuk menghadirkan sistem lelang yang lebih terstruktur dan menanggulangi masalah yang hadir pada sistem lelang *online* yang ada saat ini. Untuk itu diperlukan sistem yang handal, dapat beradaptasi, dan cepat pada pengembangannya. Salah satu metode pengembangan sistem dengan menekankan kualitas adalah *Feature-driven Development*. *Feature-driven Development* diterapkan pada proses rancang bangun sistem Auctentik agar sistem dapat dikembangkan dengan terstruktur. Rancangan sistem diimplementasikan dalam kode program dengan framework Laravel (PHP) agar adaptif dengan perubahan karena memiliki arsitektur MVC yang jelas dan package yang tersedia juga cukup banyak.

Kata kunci: *start-up*, *fdd*, *feature-driven development*, *e-commerce*, *fashion hypebeast*, *Laravel*.

GLOSARIUM

<i>Auction</i>	bisnis jual beli barang atau jasa dengan menawarkan dan mengambil tawaran, dan kemudian menjual item ke penawar tertinggi.
<i>Bid</i>	tawaran yang diajukan pada saat lelang.
<i>Debugging</i>	proses mencari dan memperbaiki masalah yang timbul pada saat pengembangan.
<i>Domain</i>	nama yang diberikan untuk mengidentifikasi sebuah jaringan tanpa menggunakan <i>internet protocol</i> (IP).
<i>Framework</i>	sebuah kerangka program yang digunakan untuk membantu developer untuk mengembangkan kode secara konsisten.
<i>Front-end</i>	tampilan yang dapat dilihat langsung oleh pengguna.
<i>Hosting</i>	tempat untuk menyimpan segala macam kebutuhan suatu website agar dapat <i>online</i> di internet.
<i>Plugin</i>	perangkat lunak tambahan untuk menjalankan fitur tambahan.
<i>Server</i>	sebuah sistem computer yang menyediakan jenis layanan tertentu dalam sebuah jaringan computer.
<i>Shortcut</i>	jalan pintas dalam melakukan perintah.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Pengembangan	2
1.5 Manfaat Pengembangan	3
1.6 Metode Pengembangan	3
1.7 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	6
2.1 <i>Start-up</i>	6
2.2 <i>E-marketplace</i> dan <i>E-commerce</i>	6
2.3 Lelang	7
2.4 Lelang Online	7
2.5 <i>Fashion Hypebeast</i>	7
2.6 <i>Feature-driven Development</i>	8
2.7 Laravel.....	10
2.8 Aktivitas Perintisan Ide Bisnis	10
BAB III ANALISIS, PERANCANGAN DAN PENGEMBANGAN.....	13
3.1 <i>Develop an Overall Model</i>	13
3.2 <i>Build a Features List</i>	16

3.3	<i>Plan by Feature</i>	23
3.4	<i>Design by Feature</i>	24
3.5	<i>Build by Feature</i>	42
BAB IV HASIL DAN PEMBAHASAN		47
4.1	Hasil.....	47
4.2	Pembahasan	58
4.2.1	Penyelesaian Masalah.....	58
4.2.2	Pengembangan Adaptif	66
4.2.3	Pengembangan Cepat	69
BAB V KESIMPULAN DAN SARAN		72
5.1	Kesimpulan.....	72
5.2	Saran	72
DAFTAR PUSTAKA		73
LAMPIRAN.....		76

DAFTAR TABEL

Tabel 2.1 Perbandingan FDD dan Extreme Programming (XP) (Arimbawa, 2014)	9
Tabel 3.1 <i>Gantt Chart</i> rencana pengembangan sistem Auctentik.	24
Tabel 4.1 <i>Gantt Chart</i> realisasi pengembangan sistem Auctentik	70

DAFTAR GAMBAR

Gambar 2.1 Lima fase <i>Feature-driven Development</i> (Coad, Lefebvre, Luca, & Luca, 1999)..9	9
Gambar 2.2 <i>Business Model Canvas</i> (BMC) ide catering yang pertama. 11	11
Gambar 2.3 <i>Business Model Canvas</i> (BMC) ide catering yang kedua..... 12	12
Gambar 2.4 <i>Business Model Canvas</i> (BMC) ide lelang <i>fashion</i> yang pertama..... 12	12
Gambar 3.1 <i>Activity Diagram</i> pelemang pada sistem lelang <i>online</i> saat ini. 13	13
Gambar 3.2 <i>Activity Diagram</i> penawar pada sistem lelang <i>online</i> saat ini. 13	13
Gambar 3.3 <i>Activity Diagram</i> pelemang pada sistem Auctentik..... 15	15
Gambar 3.4 <i>Activity Diagram</i> penawar pada sistem Auctentik..... 15	15
Gambar 3.5 <i>Use Case Diagram</i> " <i>Overall Model</i> ". 15	15
Gambar 3.6 <i>Use Case Diagram</i> set fitur <i>Authenticating a user</i> 17	17
Gambar 3.7 <i>Use Case Diagram</i> set fitur <i>Bidding an item</i> 18	18
Gambar 3.8 <i>Use Case Diagram</i> set fitur <i>Getting an item</i> 18	18
Gambar 3.9 <i>Use Case Diagram</i> set fitur <i>Managing a balance</i> 19	19
Gambar 3.10 <i>Use Case Diagram</i> set fitur <i>Managing a profile</i> 20	20
Gambar 3.11 <i>Use Case Diagram</i> set fitur <i>Authenticating a partner</i> 20	20
Gambar 3.12 <i>Use Case Diagram</i> set fitur <i>Managing an item</i> 21	21
Gambar 3.13 <i>Use Case Diagram</i> set fitur <i>Shipping an item</i> 22	22
Gambar 3.14 <i>Use Case Diagram</i> set fitur <i>Managing a balance</i> 22	22
Gambar 3.15 <i>Use Case Diagram</i> set fitur <i>Managing a profile</i> 23	23
Gambar 3.16 <i>Sequence Diagram</i> fitur <i>Sign-in the user to a system</i> 25	25
Gambar 3.17 <i>Sequence Diagram</i> fitur <i>Sign-up the user to a system</i> 25	25
Gambar 3.18 <i>Sequence Diagram</i> fitur <i>Sign-out the user of a system</i> 26	26
Gambar 3.19 <i>Sequence Diagram</i> fitur <i>Show the list of an item</i> 26	26
Gambar 3.20 <i>Sequence Diagram</i> fitur <i>Show the details of an item</i> 27	27
Gambar 3.21 <i>Sequence Diagram</i> fitur <i>Subscribe the current user to an item</i> 27	27
Gambar 3.22 <i>Sequence Diagram</i> fitur <i>Show the bids of an item</i> 28	28
Gambar 3.23 <i>Sequence Diagram</i> fitur <i>Add the bid by a current user</i> 28	28
Gambar 3.24 <i>Sequence Diagram</i> fitur <i>Complete the payment of an auction</i> 29	29
Gambar 3.25 <i>Sequence Diagram</i> fitur <i>Confirm the arrival of an item</i> 29	29
Gambar 3.26 <i>Sequence Diagram</i> fitur <i>Confirm the top-up to a balance</i> 30	30
Gambar 3.27 <i>Sequence Diagram</i> fitur <i>Request the cash-out of a balance</i> 30	30
Gambar 3.28 <i>Sequence Diagram</i> fitur <i>Show the profile of a current user</i> 31	31

Gambar 3.29 Sequence Diagram fitur <i>Edit the profile of a current user.</i>	31
Gambar 3.30 Sequence Diagram fitur <i>Edit the password of a current user.</i>	32
Gambar 3.31 Sequence Diagram fitur <i>Edit the address of a current user.</i>	32
Gambar 3.32 Sequence Diagram fitur <i>Edit the bank account of a current user.</i>	33
Gambar 3.33 Sequence Diagram fitur <i>Sign-in the partner to a system.</i>	33
Gambar 3.34 Sequence Diagram fitur <i>Sign-out the partner of a system.</i>	34
Gambar 3.35 Sequence Diagram fitur <i>Show the item list by a current partner.</i>	34
Gambar 3.36 Sequence Diagram fitur <i>Show the details of an item.</i>	35
Gambar 3.37 Sequence Diagram fitur <i>Show the bids of an item.</i>	35
Gambar 3.38 Sequence Diagram fitur <i>Add the item by a current partner.</i>	36
Gambar 3.39 Sequence Diagram fitur <i>Edit the item by a current partner.</i>	36
Gambar 3.40 Sequence Diagram fitur <i>Delete the item by a current partner.</i>	37
Gambar 3.41 Sequence Diagram fitur <i>Confirm the winner of an auction.</i>	37
Gambar 3.42 Sequence Diagram fitur <i>Confirm the shipment of an item.</i>	38
Gambar 3.43 Sequence Diagram fitur <i>Confirm the top-up for a balance.</i>	38
Gambar 3.44 Sequence Diagram fitur <i>Request the cash-out of a balance.</i>	39
Gambar 3.45 Sequence Diagram fitur <i>Show the profile of a current partner.</i>	39
Gambar 3.46 Sequence Diagram fitur <i>Edit the profile of a current partner.</i>	40
Gambar 3.47 Sequence Diagram fitur <i>Edit the password of a current partner.</i>	40
Gambar 3.48 Sequence Diagram fitur <i>Edit the address of a current partner.</i>	41
Gambar 3.49 Sequence Diagram fitur <i>Edit the bank account of a current partner.</i>	41
Gambar 3.50 Tampilan Windows Subsystem for Linux Ubuntu 20.04.	42
Gambar 3.51 Tampilan Laravel setelah instalasi.	43
Gambar 3.52 Salah satu <i>plugin</i> yang digunakan pada Microsoft Visual Studio Code.	43
Gambar 3.53 <i>Debugging</i> menggunakan Microsoft Edge.	44
Gambar 3.54 Tampilan GitLab.	45
Gambar 3.55 Tampilan Hostinger.....	45
Gambar 3.56 Tampilan Pusher.	46
Gambar 4.1 Tampilan pengguna pada <i>desktop</i>	47
Gambar 4.2 Tampilan partner pada <i>desktop</i>	48
Gambar 4.3 Tampilan admin pada <i>desktop</i>	48
Gambar 4.4 Autentikasi Partner.....	49
Gambar 4.5 Menambahkan barang yang akan dilelang.	50
Gambar 4.6 Autentikasi pengguna.....	51

Gambar 4.7 Barang yang dilelang.	52
Gambar 4.8 Mendaftar lelang.	53
Gambar 4.9 Mengikuti lelang.	54
Gambar 4.10 Pelunasan apabila memenangkan lelang.	55
Gambar 4.11 Mengirim barang yang dilelang.	56
Gambar 4.12 Konfirmasi penerimaan barang.	57
Gambar 4.13 <i>App keys</i> yang didapat setelah membuat <i>channel</i> pada Pusher	58
Gambar 4.14 Baris kode yang ditambahkan pada file <code>.env</code>	59
Gambar 4.15 Baris kode pada file <code>Bids.vue</code>	59
Gambar 4.16 Baris kode pada file <code>Form.vue</code>	60
Gambar 4.17 Baris kode pada file <code>bid.js</code>	61
Gambar 4.18 Baris kode pada file <code>show.blade.php</code>	62
Gambar 4.19 Baris kode pada file <code>BidController.php</code>	62
Gambar 4.20 Baris kode pada file <code>BidAdded.php</code>	63
Gambar 4.21 Baris kode pada file <code>routes/channel.php</code>	63
Gambar 4.22 Baris kode pada file <code>routes/web.php</code>	63
Gambar 4.23 Detail barang yang dilelang	64
Gambar 4.24 Tampilan lelang pengguna 1	64
Gambar 4.25 Tampilan lelang pengguna 2	64
Gambar 4.26 Peringatan ketika mendaftar lelang	65
Gambar 4.27 Gagal mendaftar lelang	65
Gambar 4.28 Berhasil mendaftar lelang	65
Gambar 4.29 Halaman admin untuk menambah partner	66
Gambar 4.30 Halaman <i>login</i> partner	66
Gambar 4.31 Tampilan halaman <i>developer</i> Google	67
Gambar 4.32 Baris kode pada <i>file</i> <code>.env</code>	67
Gambar 4.33 Baris kode pada <i>file</i> <code>AuthController.php</code>	68
Gambar 4.34 Baris kode pada <i>file</i> <code>routes/web.php</code>	68
Gambar 4.35 Halaman tampilan otentikasi Google	69
Gambar 4.36 Tampilan halaman <i>login</i> bawaan Laravel	70
Gambar 4.37 Tampilan halaman <i>register</i> bawaan Laravel	71

BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan *start-up* di Indonesia termasuk sangat pesat, dapat kita lihat dari data jumlah *start-up* yang ada di Indonesia sendiri. Berdasarkan situs Startup Ranking per 17 November 2019, jumlah *start-up* di Indonesia mencapai 2.164, yang menempatkan Indonesia pada posisi kelima dalam jumlah *start-up* terbanyak di dunia (StartupRanking, n.d.). Tidak menutup kemungkinan bahwa *start-up* di Indonesia akan berkembang lebih pesat lagi, menurut publikasi riset perusahaan media Catcha Group pada tahun 2025 nanti nilai pasar ekonomi digital Asia Tenggara diprediksi akan mencapai US\$ 240 miliar dengan Indonesia sebagai salah satu kontributor utama (Praditya, 2019), dan hingga Februari 2019, salah satu dari tiga sektor *start-up* dengan pertumbuhan tertinggi adalah sektor *e-commerce* (Azzura, 2019).

E-commerce di Indonesia sendiri, menurut Badan Pusat Statistik (BPS) dalam Data Sensus Ekonomi 2016 tumbuh sekitar 17% dalam sepuluh tahun terakhir dengan jumlah hingga 26,2 juta unit usaha. Seratus juta pengguna internet di Indonesia menjadi kekuatan pendorong pertumbuhan *e-commerce* (Merchant Machine, 2019). Pertumbuhan *e-commerce* di Indonesia ini juga secara tidak langsung membentuk pasar tersendiri yang biasa disebut *e-marketplace* yang mempertemukan pembeli dan penjual baik individu hingga lingkup perusahaan untuk saling berinteraksi tanpa terbatas teritori ruang dan waktu (Indrajit, 2012).

Tidak sedikit dari *e-commerce* yang ada di Indonesia bergerak pada bidang fashion, sebut saja Zalora, Zilingo, Berry Benka, Thefthing, Hijup dan lain-lain. Pasar *fashion* di Indonesia sendiri merupakan pasar yang sangat besar. Dapat dilihat juga bahwa fashion merupakan sektor *e-commerce* dengan pendapatan tertinggi yang bahkan diproyeksikan mencapai US\$ 11,7 miliar pada tahun 2023 (Jayani, 2019). Pada pasar *fashion* juga banyak yang melakukan transaksi dengan cara lelang secara daring.

Setelah mewawancarai beberapa orang yang pernah mengikuti atau mengadakan lelang barang *fashion* secara daring, didapatkan beberapa masalah pada sistem lelang *fashion* secara daring yang ada saat ini, yaitu *jump bid*, *bid & run* dan keraguan akan orisinalitas barang yang dilelang. *Jump bid* adalah keadaan dimana penawar melakukan penawaran terhadap barang yang dilelang dengan nominal yang melebihi ketentuan awal dari pelelang, sedangkan *bid &*

run adalah keadaan dimana pemenang lelang tidak menyelesaikan transaksi yang sudah dimenangkan.

Berdasarkan data dan permasalahan yang sudah dijabarkan maka dirintislah Auctentik. Auctentik sendiri adalah *start-up* yang bergerak di bidang *fashion* secara *e-commerce*, berfokus kepada penjualan dan lelang barang-barang “*hype*” dan brand lokal, dengan tujuan menjembatani masyarakat yang ingin menjual ataupun membeli barang-barang *fashion* yang tergolong “*hype*” dan atau “*limited*”. Auctentik juga bertujuan menyelesaikan masalah yang ditemukan pada sistem lelang secara daring yang ada saat ini, dan untuk dapat bersaing, Auctentik perlu dikembangkan secara cepat dan adaptif, cepat dalam masa pengembangannya dan dapat dengan mudah beradaptasi atau dilakukan ubahan sesuai dengan kondisi yang dibutuhkan.

1.2 Rumusan Masalah

Bagaimana melakukan pengembangan sistem Auctentik yang cepat dalam pengembangannya dan adaptif terhadap pengembangan-pengembangan yang akan dilakukan pada masa mendatang?

1.3 Batasan Masalah

Adapun hal-hal yang menjadi Batasan dalam pengembangan Auctentik antara lain:

- a. Sistem yang dikembangkan mengutamakan fungsionalitas utama dalam penyelesaian masalah yang ada pada sistem lelang *online* yang ada saat ini.
- b. Sistem yang dikembangkan berfokus pada fungsionalitas, sehingga sedikit mengorbankan tampilan untuk dapat mengembangkan dengan cepat.
- c. Sistem yang dikembangkan berbasis *web*, sehingga diperlukan koneksi internet untuk dapat mengaksesnya.
- d. Sistem yang dikembangkan menggunakan metode *Feature-driven Development* (FDD) untuk pengembangannya.

1.4 Tujuan Pengembangan

Tujuan dari pengembangan ini adalah mengembangkan sistem untuk *start-up* Auctentik berbasis *web* secara cepat dan adaptif. Agar Auctentik dapat segera dirilis pada tahap satu dan tetap tidak menutup untuk pengembangan-pengembangan yang akan datang.

1.5 Manfaat Pengembangan

a. Bagi Tim Gaya Sejahtera Nusantara

Manfaat yang dapat diambil oleh Tim Gaya Sejahtera Nusantara adalah dapat selesainya sistem tahap pertama, sehingga dapat segera dipresentasikan untuk menarik investor atau mengikuti inkubator, juga dapat diujicobakan kepada calon pengguna, baik penjual atau pelelang maupun pembeli, sehingga Auctentik juga mendapat *beta user* dan menambah nilai jual pada inkubator dan investor.

b. Bagi Inkubator dan Investor

Manfaat yang dapat diambil bagi inkubator dan investor adalah mereka dapat memahami bagaimana sistem bekerja, sehingga dapat lebih tertarik dengan apa yang ditawarkan Auctentik, dan menawarkan pendanaan atau inkubasi.

c. Bagi Penjual atau Pelelang

Manfaat yang dapat diambil bagi penjual atau pelelang adalah mereka dapat mencoba sistem Auctentik dari sisi penjual atau pelelang dan memahami apa yang ditawarkan oleh sistem Auctentik dibanding dengan sistem lelang *online* yang ada saat ini.

d. Bagi Pembeli atau Penawar

Manfaat yang dapat diambil bagi pembeli atau penawar adalah mereka dapat mencoba sistem Auctentik dari sisi pembeli atau penawar dan memahami apa yang ditawarkan oleh sistem Auctentik dibanding dengan sistem lelang *online* yang ada saat ini.

1.6 Metode Pengembangan

Dalam rancang bangun Auctentik, metode yang digunakan baik dalam analisa, perancangan maupun pengembangan sistem menggunakan metode *Feature-driven Development*. Yang terdiri dari lima fase berurut “yang berkolaborasi” pada proses rancang bangunnya, yaitu:

a. Mengembangkan Suatu Model yang Bersifat Keseluruhan (*Develop an Overall Model*)

Pada fase pertama, dilakukan analisis oleh tim terhadap klien, lalu bersama mendefinisikan konteks, gambaran umum, tujuan dan jangkauan perangkat lunak yang akan dikembangkan.

b. Mengembangkan Daftar Fitur-fitur (*Build a Features List*)

Fase kedua, diambil model dari hasil fase pertama untuk diidentifikasi fitur-fitur apa saja yang sesuai dengan model dan dapat dimasukkan dalam list fitur yang akan dikerjakan dalam pengembangan, dengan menggunakan bahasa yang sederhana karena akan ditujukan untuk

klien, sehingga klien dapat lebih mudah memahami apa yang bisa dilakukan oleh aktor dalam model tersebut.

c. Perencanaan Berdasarkan Fitur (*Plan by Feature*)

Pada fase ketiga, diambil semua fitur yang sudah didefinisikan, lalu di-*range* waktu pengerjaan per-fitur yang ada.

d. Perancangan Berdasarkan Fitur (*Design by Feature*)

Fase keempat, dibuat rancang bangun tiap-tiap fitur yang ada. Rancang bangun ini ditujukan untuk klien lebih memahami bagaimana perjalanan tiap fitur, dan apa yang dibutuhkan tiap fitur untuk dapat dibangun.

e. Pengembangan Berdasarkan Fitur (*Build by Feature*)

Fase terakhir adalah membangun sistem fitur demi fitur yang sudah dideklarasikan sebelumnya dan sesuai rancangan pada fase keempat. Pembangunan sistem menggunakan bahasa pemrograman dan tools yang sesuai, juga membuat interface dari sistem.

1.7 Sistematika Penulisan

Sistematika penulisan bertujuan untuk memudahkan dalam memahami isi dan maksud dari laporan tugas akhir. Berikut ini merupakan sistematika penulisan yang terdiri dari lima bab, yaitu:

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang pengembangan sistem, rumusan masalah, Batasan masalah, tujuan pengembangan, manfaat pengembangan, metode pengembangan dan sistematika penulisan terkait pengembangan sistem berbasis *web* dengan metode *Feature-driven Development (FDD)* untuk *start-up* Auctentik.

BAB II LANDASAN TEORI

Bab ini berisi landasan teori dari konsep dasar *feature-driven development*, pengertian *start-up*, definisi *e-marketplace*, pengertian *website*, definisi lelang, pengertian *fashion* dan aktivitas perintisan ide bisnis.

BAB III ANALISIS, PERANCANGAN DAN PENGEMBANGAN

Bab ini berisi uraian terkait metode yang digunakan dalam pengembangan yakni *Feature-driven Development (FDD)* beserta deskripsi dan tahapan-tahapan yang dilakukan, meliputi fase *develop an overall model*, *build a features list*, *plan by feature*, *design by feature*, dan fase *build by feature*.

BAB IV HASIL DAN PEMBAHASAN

Bab ini berisi uraian dan pembahasan terkait hasil yang dicapai beserta kelebihan dan kekurangan dalam pengembangan berdasar metode yang digunakan beserta dokumentasi atas hasil tersebut.

BAB V KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran terhadap pengembangan yang telah dilakukan untuk memberikan rangkuman terhadap hasil pengembangan sistem Auctentik, serta demi kepentingan dokumentasi atau pengembangan di masa mendatang.

BAB II

LANDASAN TEORI

2.1 Start-up

Start-up adalah perusahaan yang sedang berada dalam tahap pertama pengoperasiannya. Perusahaan-perusahaan ini mulanya sering didanai oleh para wirausahawan yang mendirikan perusahaan sementara mereka mencoba untuk mengkapitalisasi produk maupun servis yang dikembangkan yang mereka percayai dapat menjadi permintaan (Grant, 2020). Menurut Neil Blumenthal, *start-up* adalah perusahaan yang bekerja untuk menyelesaikan suatu masalah yang mana solusinya masih belum jelas dan dengan tingkat kesuksesan yang tidak terjamin. Menurut Adora Cheung, *start-up* adalah sebuah keadaan pikiran, dimana orang-orang masuk ke dalam perusahaan dan tetap membuat keputusan yang tegas untuk meninggalkan kebiasaan yang sudah stabil sebagai penukaran janji dari mendapatkan pertumbuhan yang luar biasa serta adanya kegembiraan dalam membuat dampak yang bisa dirasakan secara langsung. Menurut The American Heritage Dictionary, *start-up* adalah bisnis atau usaha yang baru mulai beroperasi (Robehmed, 2013).

2.2 E-marketplace dan E-commerce

Penggunaan internet dalam transaksi elektronik ini berdampak sangat positif dalam interaksi secara global tanpa batasan tempat dan waktu. Transaksi dapat dilakukan dengan lebih praktis tanpa memerlukan perjanjian tatap muka antar pelaku bisnis, sehingga dapat dikatakan perdagangan elektronik atau *e-commerce* ini menjadi penggerak ekonomi baru dalam bidang teknologi khususnya di Indonesia (Dr. Halim Barkatullah, 2017).

E-marketplace adalah pasar elektronik berbasis internet yang memungkinkan komunikasi dan transaksi bisnis-ke-bisnis secara daring (Liberati, 2009). *E-marketplace* adalah suatu bentuk melakukan bisnis yang menggunakan teknologi internet untuk menyatukan banyak *vendor* dan perusahaan pelanggan di satu situs *web* atau platform untuk melakukan perdagangan (Janita, 2016).

E-commerce adalah pembelian dan penjualan barang dan jasa melalui internet (Wang, Yao, & Xing, 2016). *E-commerce* adalah berbagai transaksi dan interaksi komersial elektronik yang dilakukan melalui internet, situs *e-commerce* biasanya menyediakan layanan satu-ke-banyak (David-West, 2016).

Dengan kata lain e-marketplace adalah sebuah tempat yang terdapat banyak *supplier* atau produsen yang menjual barangnya secara online, sehingga konsumen dapat membeli dari beberapa penjual yang berbeda, sedangkan e-commerce hanya terdapat satu *supplier* yang menjual barang secara online (Maulana, Susanto, & K., 2019).

2.3 Lelang

Kata lelang diambil dari kata *Auctio*, yang artinya peningkatan secara bertahap. Berbeda dengan proses jual-beli, lelang adalah penjualan umum atau penjualan barang-barang yang dilakukan kepada umum dengan harga penawaran yang meningkat atau menurun atau dengan memasukkan harga dalam sampul tertutup, atau dijual kepada orang yang menyetujui harga yang ditawarkan atau memasukkan harga dalam sampul tertutup (Risman, 2015).

Lelang menggunakan penetapan harga dinamis tergantung pada nilai penawaran (Bubendorfer, Palmer, & Welch, 2009). Lelang adalah mekanisme yang dijelaskan oleh seperangkat aturan lelang yang menentukan bagaimana pemenang ditentukan dan berapa yang harus dia bayar (Anthony, 2009).

Sebagaimana dimaksud dalam Pasal 1 Sub 17 Undang-Undang Nomor 19 Tahun 2000 tentang Penagihan Pajak Dengan Surat Paksa dijelaskan bahwa lelang adalah penjualan barang di muka umum dengan cara penawaran harga secara lisan dan atau tertulis melalui usaha pengumpulan peminat atau calon pembeli.

2.4 Lelang Online

Lelang *online* adalah versi lelang berbasis internet dari lelang tradisional dimana tidak terbatas waktu dan lokasi (Anthony, 2009). Lelang *online* adalah model yang digunakan untuk membuat bisnis dimana peserta menawar untuk produk dan layanan melalui internet (Peterson, 2008).

2.5 Fashion Hypebeast

Fashion adalah jenis pakaian, aksesoris dan ornamen yang digunakan atau menjadi kebiasaan selama periode waktu tertentu (Sánchez & Sánchez, 2020). *Fashion* adalah ekspresi status sosial, gaya hidup dan rasa melalui penggunaan benda oleh individu (Ayertey & Howell, 2019).

Istilah Hypebeast paling sering digunakan sebagai bahasa gaul yang menggambarkan seseorang yang merupakan *beast* atau binatang buas (terobsesi) tentang merek-merek fashion

yang trendi dan dengan demikian akan melakukan cara apa pun untuk mencapai hype itu (Esquireme, n.d.). 'Beast' biasanya mengacu pada Milenial dan Gen-X yang kaya dan ingin menghabiskan pendapatan mereka untuk barang-barang fashion streetwear yang mahal dan mahal yang telah digunakan oleh influencer terkemuka dari komunitas Hypebeast dan dipopulerkan oleh selebriti seperti Kanye West dan Jay-Z di media sosial. Merek streetwear yang telah menerima cap persetujuan dari komunitas Hypebeast biasanya menumbuhkan 'hype' tertentu dengan hanya menjual persediaan eceran terbatas (Orloff & Gwyn, 2017).

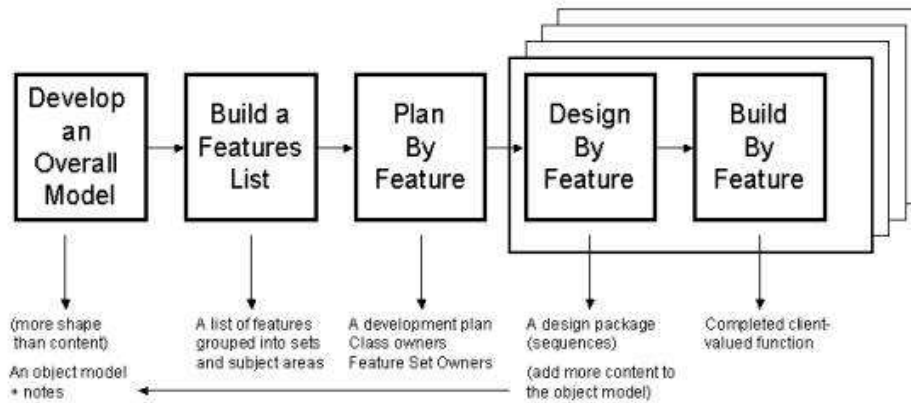
Fenomena Hypebeast tentunya telah mendominasi gaya hidup anak muda secara global; pengikutnya yang paling setia telah mendapatkan reputasi sebagai pemanjat sosial yang bekerja tanpa lelah untuk mengembangkan citra mereka di media sosial dengan memamerkan pakaian kelas atas mereka.

2.6 Feature-driven Development

Feature-driven Development adalah proses pengembangan perangkat lunak yang cepat, sangat adaptif terhadap perkembangan, yang mana menekankan pada kualitas di tiap-tiap langkahnya, memberikan hasil yang sering dan nyata pada tiap langkahnya, memberikan kemajuan dan informasi yang akurat dan bermakna, serta disukai oleh klien, manajer dan pengembang (Goyal, 2008).

Metode *Feature-driven Development* sendiri konsepnya dikemukakan oleh Peter Coad sebagai model proses praktis untuk rekayasa perangkat lunak berorientasi objek (Coad, Lefebvre, Luca, & Luca, 1999), yang lalu diadaptasi oleh Stephen Palmer dan Felsing John sehingga proses dapat lebih adaptif, cepat, dan dapat diterapkan pada pengembangan perangkat lunak dengan skala yang lebih besar (Palmer & Felsing, 2002). Menghasilkan tiga poin penting yaitu:

- a. Menitikberatkan pada kerjasama dan kolaborasi antar individu yang berada dalam tim.
- b. Manajemen kompleksitas perangkat lunak, dengan membagi pengembangan berdasarkan fitur yang diikuti dengan integrasi perangkat lunak.
- c. Menjalankan komunikasi teknis rinci secara intensif menggunakan sarana verbal, visual, ataupun teks (Palmer & Felsing, 2002).



Gambar 2.1 Lima fase *Feature-driven Development* (Coad, Lefebvre, Luca, & Luca, 1999).

Metode *Feature-driven Development* terdiri dari lima fase berurut pada proses rancang bangunnya (Gambar 2.1). Pada fase pertama, klien bersama tim pengembang bekerja sama untuk mengembangkan keseluruhan model dari domain klien. Klien menyajikan langkah-langkah yang sesuai konteks dan sistem yang diperlukan. Sementara tim pengembang membuat model dari informasi ini. Proses menyesuaikan bentuk model menghasilkan model final yang disepakati. Pada fase kedua, tim memformulasikan informasi yang dikumpulkan dari fase pertama melalui pembuatan list fitur. Fitur adalah fungsi atau keluaran kecil yang dapat dinilai oleh klien. Tim membuat daftar fitur dengan mengidentifikasi fitur, memprioritaskannya sesuai dengan kepuasan klien dan menimbanginya untuk memastikan fitur tersebut dapat diselesaikan dalam kurun waktu dua minggu atau kurang. Fitur juga dikelompokkan bersama ke dalam set fitur agar dapat dilihat ketergantungan dan asosiasi antar fitur. Fase ketiga melibatkan penggunaan fitur untuk merencanakan pengembangan dan mengatur tim di persiapan untuk implementasi. Dua fase terakhir adalah iteratif dan setiap iterasi berlangsung 2 minggu atau kurang. Manfaat utama dari ini adalah memastikan proyek diselesaikan dalam jangka waktu yang sesuai (Feature Driven Development & Empirical Modelling, 2008).

Tabel 2.1 Perbandingan FDD dan Extreme Programming (XP) (Arimbawa, 2014)

Pembanding	FDD	XP
Sifat developer	Menekankan hirarki proses	Peer-to-peer
Kepemilikan proses	Individual, masing-masing memiliki Class owners	Kolektif
Sukses jika	Didukung developer yang di atas rata-rata	Didukung developer yang rata-rata

Peran serta Client	Hanya pada proses 1, 2, dan 4	Client ikut dalam pengerjaan
Perubahan dapat dilakukan pada	Proses 1	Konstant pada setiap proses
Jumlah jam kerja	Bebas	40jam/minggu

Herarki dalam tim pengembang pada FDD memungkinkan untuk menjaga Tim baik skala kecil maupun besar. Namun karena FDD menggunakan sifat individual dari kepemilikan proses, maka dapat terjadi saling tunggu antara feature yang terkait. Perubahan feature hanya dapat dilakukan hanya pada proses pertama dan secara keseluruhan hanya mampu memberikan penambahan kurang dari 10%. Dibandingkan dengan XP, FDD lebih banyak memberikan pemodelan daripada langsung melakukan pengkodean membuat hasil akan terlihat dalam waktu yang lama, walaupun hal ini memberikan keuntungan untuk proyek dengan skala dan kerumitan lebih besar.

2.7 Laravel

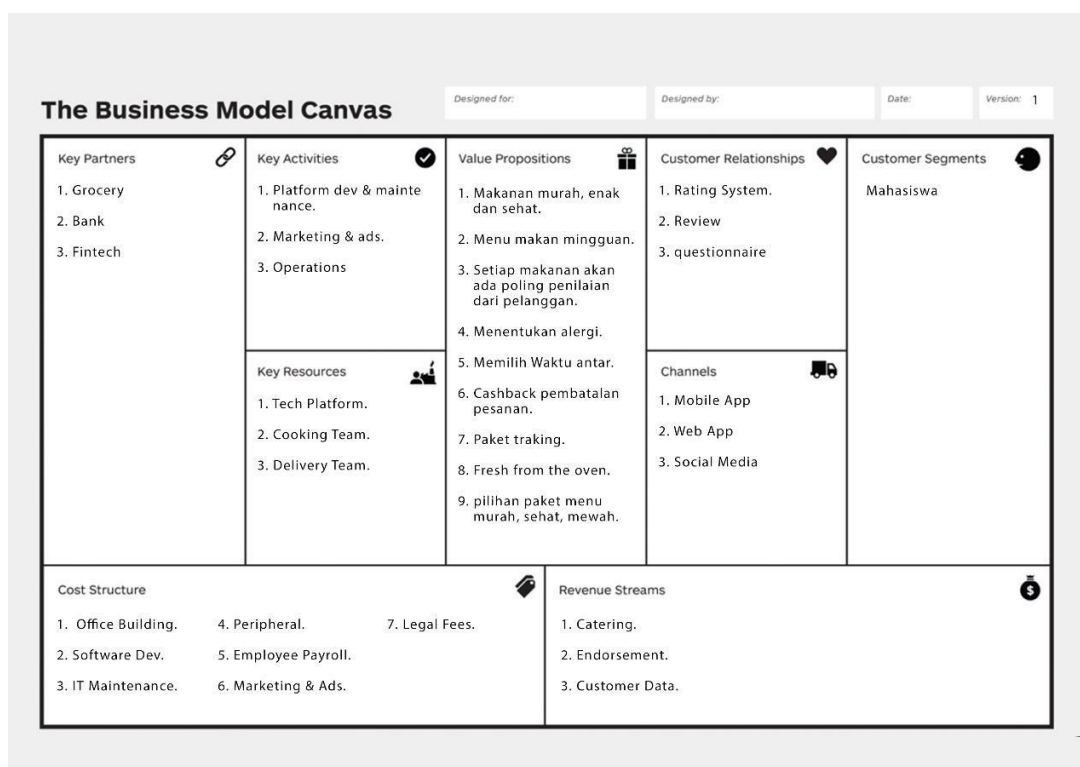
Laravel adalah framework aplikasi web, open source, dan digunakan secara luas untuk perancangan aplikasi web yang cepat dan mudah. Laravel dibuat oleh Taylor Otwell pada tahun 2011. Framework ini dibuat untuk pengembangan aplikasi website dimana mengikuti arsitektur MVC (Model View Controller). MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti: manipulasi data, controller, dan user interface (Hanry Ham, 2018).

Sintaks Laravel ekspresif, jelas, dan menghemat waktu. Laravel mempermudah proses pengembangan web dengan banyaknya *Package*, sehingga mempermudah tugas-tugas yang umum seperti *routing*, *authentication*, *sessions*, dan *caching*.

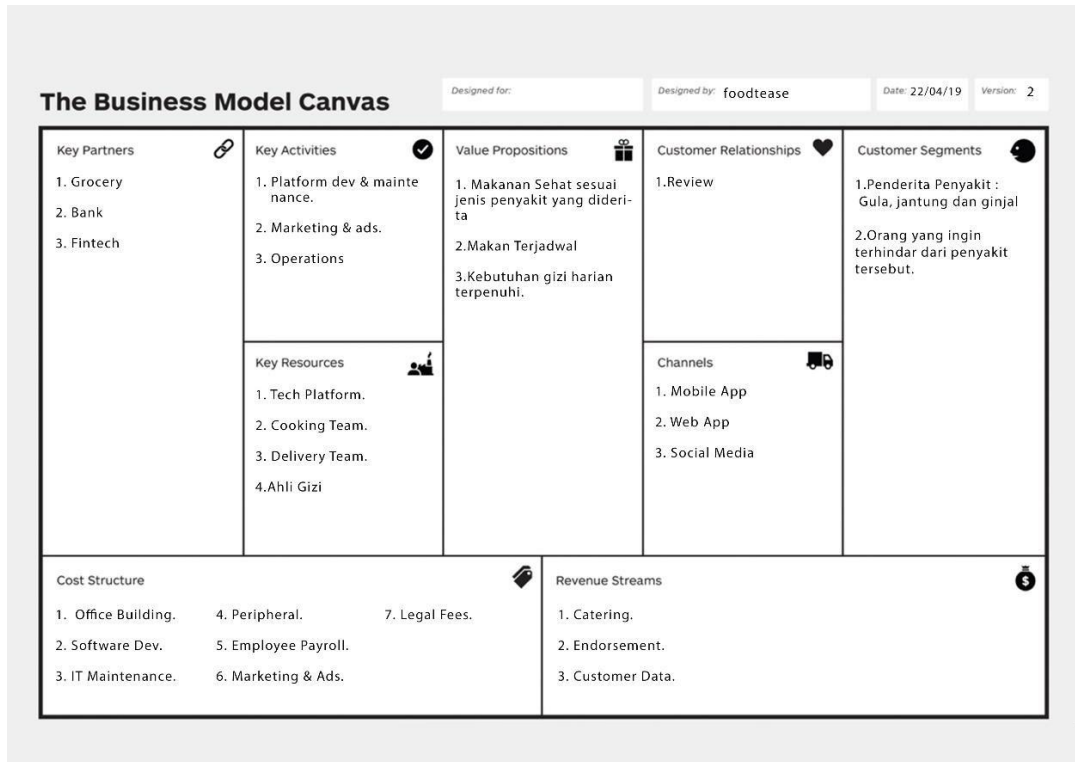
2.8 Aktivitas Perintisan Ide Bisnis

Anggota kelompok kami berangkat dengan ide masing-masing ketika masuk kelas bisnis. Fendi dengan ide voucher diskon yang terintegrasi pada satu sistem dari beberapa *E-marketplace*, Rifqi dengan ide integrasi tukang sayur keliling dalam satu sistem dan saya dengan ide bengkel kendaraan bermotor yang berintegrasi pada sebuah sistem. Begitu menyampaikan ketiga ide tersebut kepada Pak Hari kami mendapat masukan masing-masing. Setelah itu Pak Hari memberi sebuah ide tentang catering *online*. Kamipun membicarakannya dan tertarik untuk mengerjakan ide catering tersebut. Setelah itu kami membuat *Business Model Canvas* (BMC)(Gambar 2.2), *Market Research* dan *prototype*-nya. Lalu pada minggu

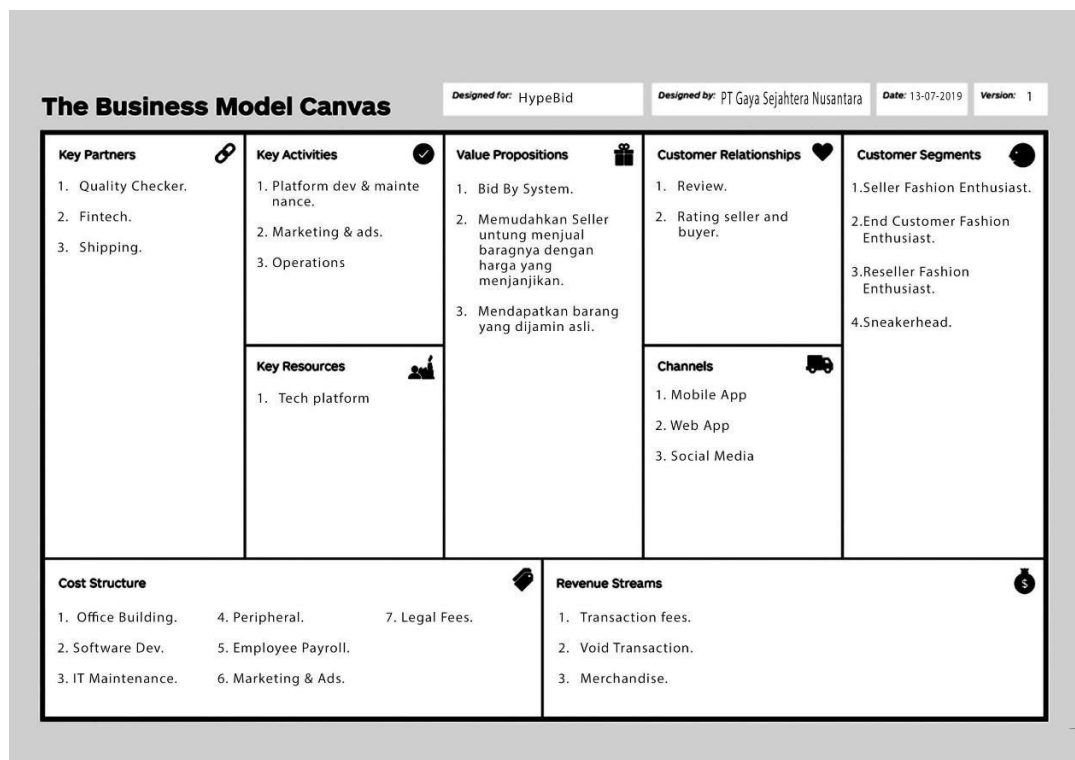
selanjutnya kami mempresentasikan ide tersebut dengan BMC yang sudah direvisi (Gambar 2.3) kepada Pak Arif dan Pak Bagus perwakilan dari IBISMA. Disini kami mendapat pencerahan bahwasanya dengan target *customer* mahasiswa tidak banyak yang bisa diharapkan dengan ide catering. Setelah acara di cohive kami berkumpul dan mumutar ide lagi, mencari pasar yang lebih menggiurkan untuk dieksekusi sebagai ide bisnis. Ada sebuah ide catering tentang makanan sehat terlintas. Tapi ada ide yang lebih menggiurkan yaitu sebuah aplikasi lelang barang-barang *fashion hypebeast* dengan *customer* yang loyal untuk mengeluarkan uang dan pasar yang besar di Indonesia, dan dibuatlah BMC untuk ide ini (Gambar 2.4).



Gambar 2.2 *Business Model Canvas* (BMC) ide catering yang pertama.



Gambar 2.3 Business Model Canvas (BMC) ide catering yang kedua.



Gambar 2.4 Business Model Canvas (BMC) ide lelang fashion yang pertama.

BAB III

ANALISIS, PERANCANGAN DAN PENGEMBANGAN

3.1 Develop an Overall Model

Pada fase pertama, saya, Fendi dan Rifqi sebagai tim pengembang dan juga Rifqi yang merangkap sebagai klien karena pernah mengikuti lelang *online* yang ada di platform Instagram, kami berdiskusi menganalisa sistem lelang *online* yang ada saat ini, dengan pertama kami me-*breakdown* proses bisnis pada lelang *online* yang ada saat ini, baik dari sudut pandang pelelang maupun penawar barang. Sehingga dihasilkan diagram aktivitas pelelang barang seperti pada Gambar 3.1 dan penawar barang seperti pada Gambar 3.2.



Gambar 3.1 Activity Diagram pelelang pada sistem lelang *online* saat ini.



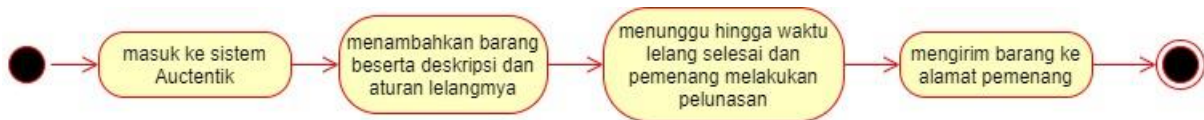
Gambar 3.2 Activity Diagram penawar pada sistem lelang *online* saat ini.

Setelah me-*breakdown* proses bisnis, dapat kami temukan beberapa masalah yang timbul pada sistem lelang online yang ada saat ini. Diantaranya adalah *jump bid*, *bid & run* dan *doubt of authenticity*. *Jump bid* adalah keadaan dimana pada sistem lelang yang ada saat ini, penawar melakukan penawaran melalui kolom komentar, baik itu yang ada pada platform Instagram maupun grup Facebook, sehingga penawar bebas memasukkan nominal yang ingin dia tawarkan, menghiraukan aturan yang biasanya sudah disebutkan oleh pelelang perihal kelipatan *bid* atau tambahan nominal ketika akan menawar, yang mana kadang menghasilkan

lompatan tawaran yang cukup tinggi, sehingga kegiatan lelang online menjadi kurang kompetitif. Kadang beberapa pelelang membuat akun untuk ikut menawar dan meninggikan nominal tawaran agar dapat paling tidak mendapatkan nominal yang sesuai dengan ekspektasi pelelang, dan kebanyakan membuat nominal tawaran menjadi sangat tinggi. Lalu *bid & run*, adalah keadaan dimana pemenang lelang sudah ditetapkan, namun ketika dihubungi lebih lanjut untuk pembayaran dan konfirmasi pengiriman, pemenang lelang tersebut malah menghilang atau tidak dapat dihubungi. Sedangkan *doubt of authenticity* adalah keadaan dimana calon penawar ragu akan ke-orisinalitas-an barang yang akan dilelang, dan memang keraguan adalah masalah utama ketika membicarakan sebuah transaksi yang terjadi secara online atau tidak secara tatap muka langsung.

Dari ketiga permasalahan yang kami temukan tersebut, kami berdiskusi untuk menawarkan solusinya, dan menghasilkan beberapa solusi. Untuk *jump bid*, dibuatkanlah *bid by system*, penawar tidak perlu memasukkan nominal penawaran untuk ditawarkan, nominal penawaran akan otomatis terisi dengan nominal penawaran awal atau nominal penawaran sebelumnya ditambah dengan kelipatan penawaran, sehingga penawar hanya perlu menekan tombol untuk melakukan penawaran, juga akan membuat lelang menjadi lebih kompetitif karena nominal penawaran tidak akan melambung jauh dari nominal penawaran sebelumnya, hanya akan bertambah sesuai dengan kelipatan penawaran. Ini juga dapat menjadi solusi perihal pelelang membuat akun untuk mengikuti lelang dan menaikkan nominal penawaran, karena disini juga dibuat untuk dapat mengikuti lelang, paling tidak penawar harus memiliki deposit sebesar nominal awal penawaran atau *open bid*, sehingga untuk pelelang membuat akun dan mengikuti lelang untuk menaikkan nominal penawaran hanya akan merugikan pihak pelelang sendiri. Untuk *bid & run*, seperti yang dijelaskan sebelumnya, untuk dapat mengikuti lelang penawar harus mendepositkan dahulu uang sesuai nominal penawaran awal atau *open bid*, apabila penawar tidak memenangkan lelang, maka uang tersebut akan dikembalikan, dan apabila penawar memenangkan lelang, namun melakukan praktik *bid & run*, maka uang tersebut tidak akan dikembalikan dan menjadi sebuah *punishment* karena melakukan praktik *bid & run*. Untuk *doubt of authenticity*, kami menawarkan 100% uang kembali apabila ternyata barang yang diterima tidak autentik.

Setelah itu kami berdiskusi merancang proses bisnis baru yang akan diimplementasikan pada sistem Auctentik sesuai dengan solusi yang kami tawarkan sebelumnya, dan menghasilkan diagram aktivitas untuk proses bisnis pelelang seperti pada Gambar 3.3 dan untuk proses bisnis penawar seperti pada Gambar 3.4.

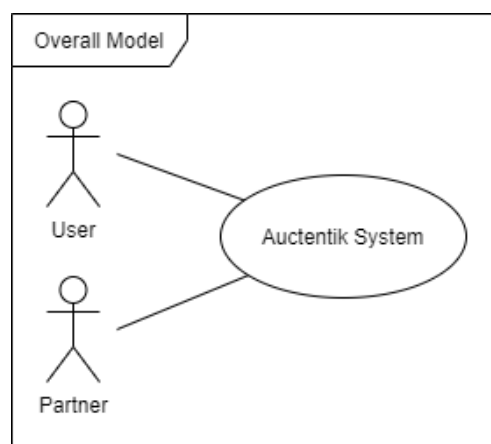


Gambar 3.3 *Activity Diagram* pelelang pada sistem Auctentik.



Gambar 3.4 *Activity Diagram* penawar pada sistem Auctentik.

Setelah menentukan proses bisnis baru, kami menentukan dan mendefinisikan model secara keseluruhan untuk sistem Auctentik ini, dan digambarkan dengan diagram use case seperti pada Gambar 3.5. Digambarkan akan ada dua aktor yang akan menggunakan sistem Auctentik yaitu user dan partner. User disini adalah penawar barang, sedangkan partner adalah pelelang yang mana untuk tahap awal kami akan menggandeng beberapa pelelang yang sudah memiliki reputasi yang bagus baik dari platform Instagram maupun facebook, sehingga juga dapat meminimalisir ketidakpuasan penawar terhadap salah satu pelelang yang ada pada sistem.



Gambar 3.5 *Use Case Diagram "Overall Model"*.

3.2 *Build a Features List*

Berdasarkan dari data-data yang sudah didapat pada fase *Develop an Overall Model* baik proses bisnis maupun model secara keseluruhan, kami pun berdiskusi membuat daftar fitur-fitur yang akan dibuat pada sistem Auctentik ini. Pada konteks pengembangan dikendalikan fitur, fitur adalah fungsi yang kelak akan dinilai oleh klien yang dapat diimplementasikan dalam dua minggu atau kurang (Coad, Lefebvre, Luca, & Luca, 1999). Dan disarankan untuk menggunakan templat penamaan untuk mendefinisikan fitur (3.1).

$$\langle \text{action} \rangle \text{ the } \langle \text{result} \rangle \langle \text{by|for|of|to} \rangle \text{ a(n) } \langle \text{object} \rangle \quad (3.1)$$

(Coad, Lefebvre, Luca, & Luca, 1999)

Fitur juga dapat dikelompokkan berdasarkan keterkaitan dengan bisnisnya dan dapat menggunakan templat penamaan set fitur (3.2).

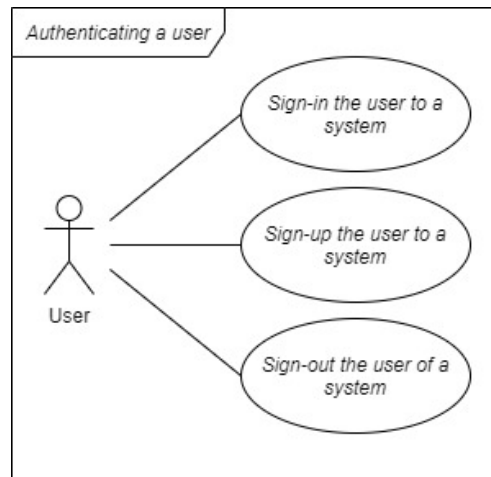
$$\langle \text{action} \rangle \langle \text{-ing} \rangle \text{ a(n) } \langle \text{object} \rangle \quad (3.2)$$

(Coad, Lefebvre, Luca, & Luca, 1999)

Berdasarkan data dan templat penamaan tersebut kami modelkan fitur-fitur menggunakan diagram use case, dan dikelompokkan berdasarkan aktor dan set fitur.

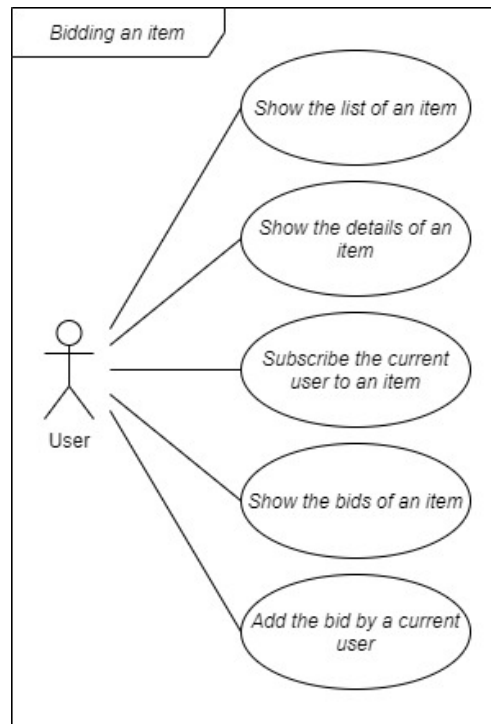
A. Aktor User

Set fitur pertama pada aktor user (Gambar 3.6) adalah autentikasi user, disini akan ada tiga fitur yang akan dibangun. Pertama adalah *Sign-in the user to a system* atau untuk memasukkan user ke dalam sistem Auctentik. Lalu *Sign-up the user to a system* atau untuk mendaftarkan user ke dalam sistem Auctentik. Terakhir adalah *Sign-out the user to a system* atau untuk mengeluarkan user dari sistem Auctentik.



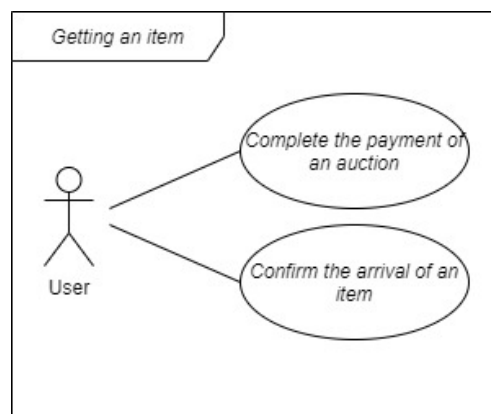
Gambar 3.6 Use Case Diagram set fitur *Authenticating a user*.

Set fitur berikutnya (Gambar 3.7) adalah menawar sebuah barang, dan akan ada lima fitur yang dibangun. Dengan fitur pertama *Show the item list of an item* atau untuk menampilkan daftar barang yang ada pada sistem. Lalu *Show the details of an item* atau untuk menampilkan detail dari suatu barang yang ditawarkan, termasuk aturan-aturan lelangnya seperti tanggal dan waktu mulai lelang, tanggal dan waktu berakhirnya lelang, harga *open bid* atau harga mulai lelang, dan *multiple bid* atau kelipatan penawaran pada lelang. Fitur berikutnya *Subscribe the current user to an item* atau untuk mendaftarkan pengguna yang masuk ke sistem pada sebuah barang yang akan dilelang. Fitur keempat *Show the bids of an item* atau untuk menampilkan tawaran-tawaran dari barang yang dilelang baik dari pengguna yang masuk ke sistem maupun pengguna lainnya. Fitur terakhir *Add the bid by a current user* atau untuk menambahkan tawaran dari pengguna yang sedang masuk ke sistem ke dalam daftar tawaran pada barang yang ditawarkan.



Gambar 3.7 Use Case Diagram set fitur *Bidding an item*.

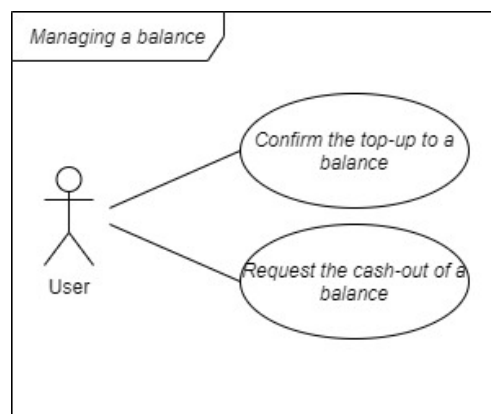
Set fitur ketiga (Gambar 3.8) adalah mendapatkan barang yang dimenangkan dalam lelang, dengan dua fitur yang akan dibangun. Fitur pertamanya *Complete the payment of an auction* atau untuk melunasi atau melengkapi pembayaran dari lelang yang dimenangkan. Lalu *Confirm the arrival of an item* atau untuk mengkonfirmasi kedatangan barang yang dikirimkan dari lelang yang dimenangkan oleh pengguna.



Gambar 3.8 Use Case Diagram set fitur *Getting an item*.

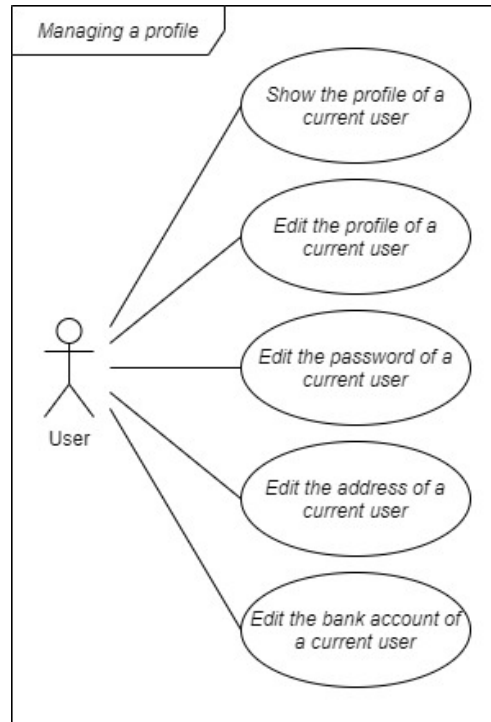
Set fitur berikutnya (Gambar 3.9) adalah mengelola saldo pengguna, dengan dua fitur yang akan dibangun. Pertama, fitur *Confirm the top-up to a balance* atau untuk mengkonfirmasi

apabila pengguna melakukan isi saldo akun pengguna. Lalu *Request the cash-out of a balance* atau untuk melakukan permohonan penarikan saldo pada akun pengguna.



Gambar 3.9 Use Case Diagram set fitur *Managing a balance*.

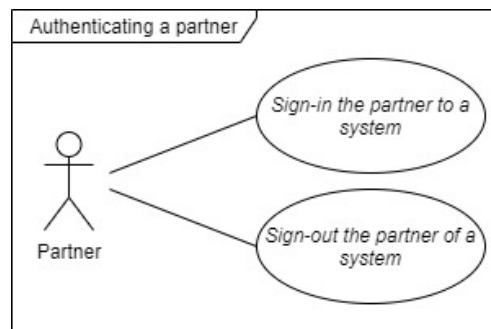
Set fitur terakhir pada aktor *user* (Gambar 3.10) adalah mengelola profil pengguna, berisi lima fitur yang akan dibangun. Fitur pertama *Show the profile of a current user* atau untuk menampilkan informasi profil dari pengguna yang sedang masuk ke sistem. Lalu *Edit the profile of a current user* atau untuk melakukan ubahan pada profil pengguna yang sedang masuk ke sistem. Fitur ketiga *Edit the password of a current user* atau untuk melakukan ubahan pada kata sandi dari pengguna yang sedang masuk ke sistem. Selanjutnya *Edit the address of a current user* atau untuk melakukan ubahan pada alamat pengguna yang sedang masuk ke sistem. Terakhir *Edit the bank account of a current user* atau untuk melakukan ubahan pada rekening bank pengguna yang sedang masuk ke sistem.



Gambar 3.10 Use Case Diagram set fitur *Managing a profile*

B. Aktor Partner

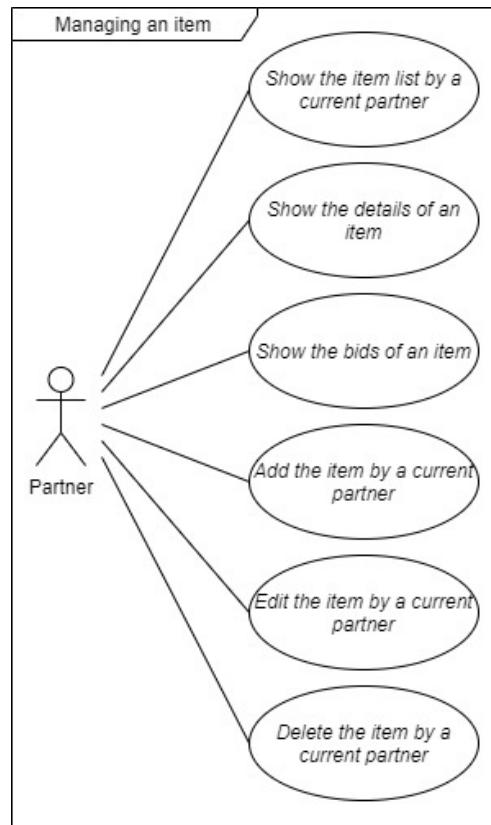
Set fitur pertama pada aktor partner (Gambar 3.11) adalah autentikasi partner, dengan dua fitur yang akan dibangun. Fitur pertama *Sign-in the partner to a sistem* atau untuk memasukkan partner ke dalam sistem Auctentik. Dan *Sign-out the partner of a system* atau untuk mengeluarkan partner dari sistem Auctentik.



Gambar 3.11 Use Case Diagram set fitur *Authenticating a partner*.

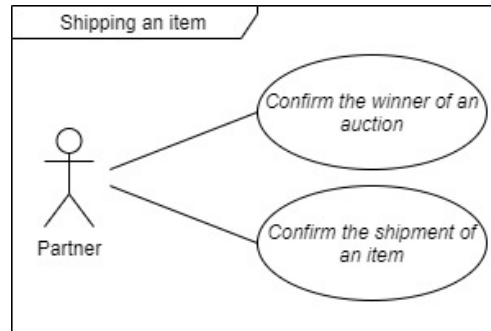
Set fitur berikutnya (Gambar 3.12) adalah mengelola barang, berisi enam fitur yang akan dibangun. Pertama *Show the item list by a current partner* atau untuk menampilkan daftar barang yang ditawarkan oleh partner yang sedang masuk ke sistem. Lalu *Show the details of an item* atau untuk menampilkan detail dari suatu barang yang ditawarkan oleh partner yang

sedang masuk ke sistem. Fitur ketiga *show the bids of an item* atau untuk untuk menampilkan tawaran-tawaran dari barang yang dilelang. Selanjutnya *add the item by a current partner* atau untuk menambahkan barang lelang dari partner yang sedang masuk ke sistem. Fitur kelima *edit the item by a current partner* atau untuk melakukan ubahan pada barang lelang dari partner yang sedang masuk ke sistem. Terakhir, *delete the item by a current partner* atau untuk menghapus barang lelang atau membatalkan lelang dari partner yang sedang masuk ke sistem.



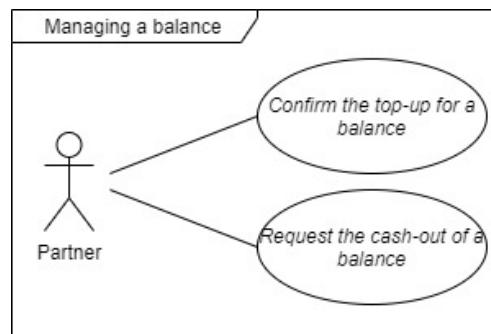
Gambar 3.12 Use Case Diagram set fitur *Managing an item*.

Set fitur ketiga (Gambar 3.13) adalah mengirim barang yang masa lelangnya sudah selesai dan dimenangkan oleh salah seorang pengguna, terdiri dari dua fitur yang akan dikembangkan. Fitur pertama *confirm the winner of an auction* atau untuk mengkonfirmasi pemenang dari lelang barang tersebut. Dan *confirm the shipment of an item* atau untuk mengkonfirmasi pengiriman barang ke alamat pengguna yang sudah memenangkan lelang barang tersebut.



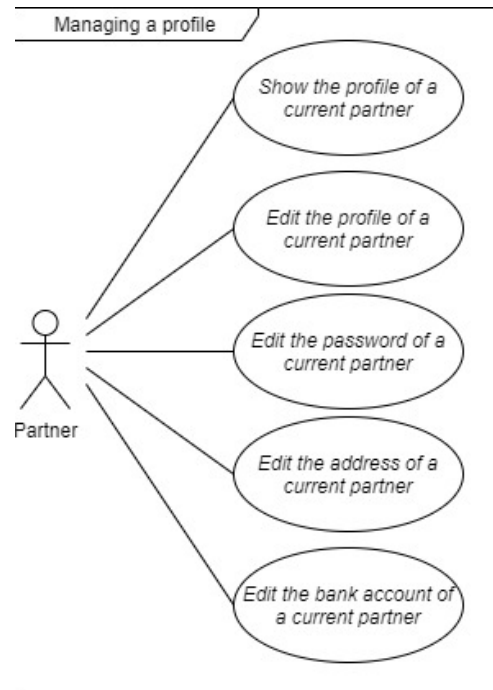
Gambar 3.13 Use Case Diagram set fitur *Shipping an item*.

Set fitur berikutnya (Gambar 3.14) adalah mengelola saldo partner, dengan dua fitur yang akan dibangun. Pertama, fitur *Confirm the top-up for a balance* atau untuk mengkonfirmasi apabila partner melakukan isi saldo akun partner. Lalu *Request the cash-out of a balance* atau untuk melakukan permohonan penarikan saldo pada akun partner.



Gambar 3.14 Use Case Diagram set fitur *Managing a balance*.

Set fitur terakhir (Gambar 3.15) adalah mengelola profil partner, berisi lima fitur yang akan dibangun. Fitur pertama *show the profile of a current partner* atau untuk menampilkan informasi profil dari partner yang sedang masuk ke sistem. Lalu *Edit the profile of a current partner* atau untuk melakukan ubahan pada profil partner yang sedang masuk ke sistem. Fitur ketiga *Edit the password of a current partner* atau untuk melakukan ubahan pada kata sandi dari partner yang sedang masuk ke sistem. Selanjutnya *Edit the address of a current partner* atau untuk melakukan ubahan pada alamat partner yang sedang masuk ke sistem. Terakhir *Edit the bank account of a current partner* atau untuk melakukan ubahan pada rekening bank partner yang sedang masuk ke sistem.



Gambar 3.15 Use Case Diagram set fitur *Managing a profile*.

3.3 Plan by Feature

Setelah mendefinisikan daftar fitur yang ada pada sistem secara keseluruhan seperti pada LAMPIRAN A, kami melakukan perencanaan pengembangan. Diambil tiap-tiap fitur dan diberikan durasi pengerjaan masing-masing fitur satu hari kerja, sehingga didapat estimasi pengerjaan 34 hari kerja. Lalu dipotong dengan libur pada tanggal merah. Perencanaan penjadwalan kami modelkan dengan *Gantt Chart* seperti pada Tabel 3.1 untuk model yang dikelompokkan berdasarkan aktor dan set fitur yang sudah didefinisikan pada fase *Build a Features List*. Untuk pemodelan yang lebih detail per-fiturnya dapat dilihat pada LAMPIRAN B.

Tabel 3.1 *Gantt Chart* rencana pengembangan sistem Auctentik.

#	Fitur	Durasi	Maret 2020			April 2020			Mei 2020		
			18-20	23,24,26,27	30-3	6-9	13-17	20-24	27-30	4-6 & 8	
1	User	17 hari									
1.1	Authenticating a user	3 hari									
1.2	Bidding an item	5 hari									
1.3	Getting an item	2 hari									
1.4	Managing a balance	2 hari									
1.5	Managing a profile	5 hari									
2	Partner	17 hari									
2.1	Authenticating a partner	2 hari									
2.2	Managing an item	6 hari									
2.3	Shipping an item	2 hari									
2.4	Managing a balance	2 hari									
2.5	Managing a profile	5 hari									

3.4 Design by Feature

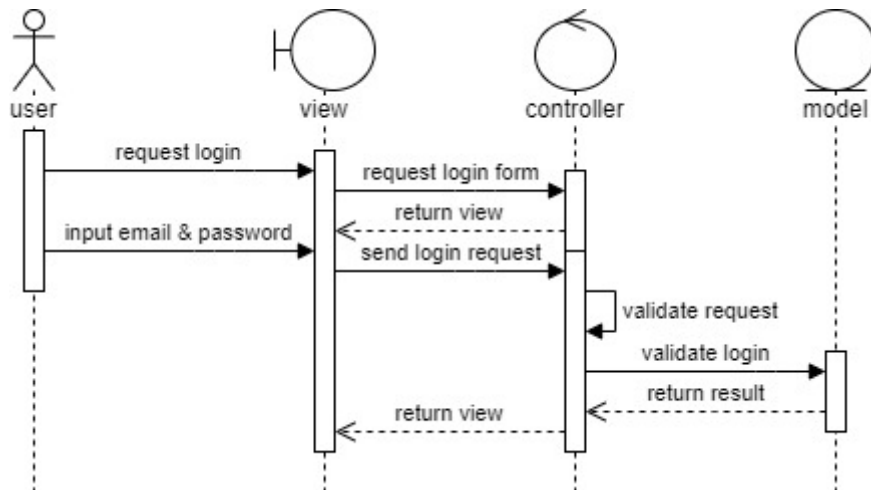
Pada fase keempat, kami mengambil tiap-tiap fitur yang didefinisikan pada fase *Build a Features List* lalu membuat rancang bangun dengan tujuan untuk klien. Rancang bangun kami kelompokkan berdasarkan aktor dan set fitur-nya lalu kami modelkan dengan *Sequence Diagram* untuk menggambarkan langkah demi langkah bagaimana sistem bekerja pada tiap-tiap fiturnya.

A. Aktor User

Authenticating a user

a. Sign-in the user to a system

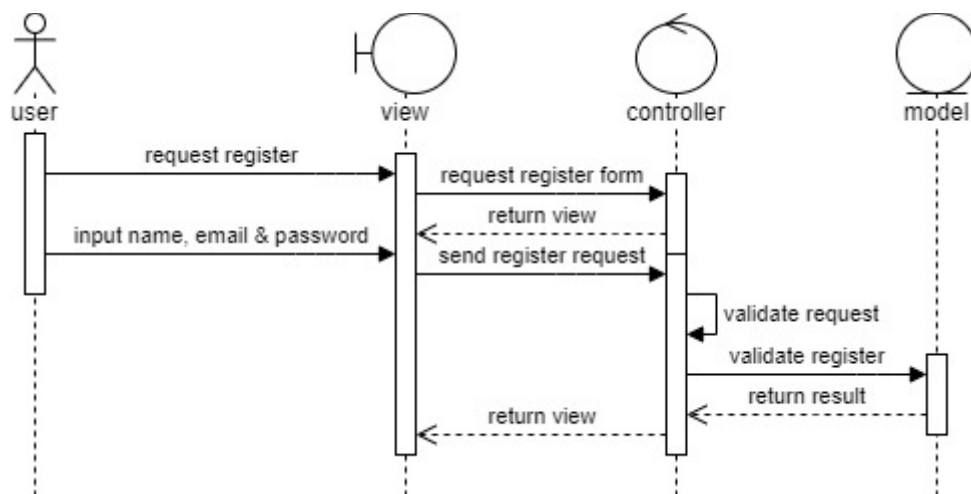
Pada fitur ini dilakukan autentikasi user untuk dapat masuk ke dalam sistem. User yang sudah memasukkan *email* dan *password*-nya kemudian divalidasi terlebih dahulu masukkannya di *controller*, lalu divalidasi lagi terhadap model User untuk dicek apakah masukan dari user sama dengan data yang disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.16.



Gambar 3.16 Sequence Diagram fitur Sign-in the user to a system.

b. Sign-up the user to a system

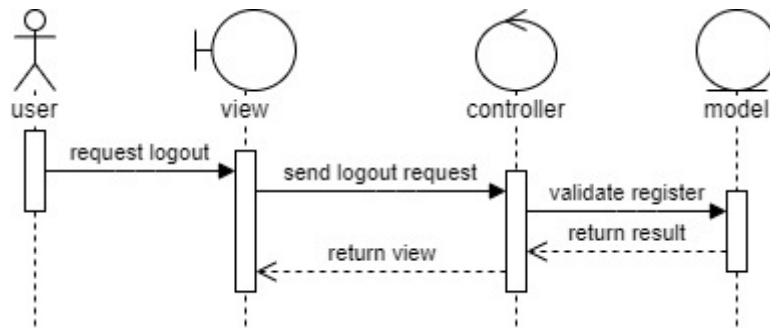
Pada fitur ini dilakukan autentikasi user untuk terdaftar ke dalam sistem. User akan memasukkan *email* dan *password*-nya, beserta namanya kemudian divalidasi terlebih dahulu masukkannya di *controller*, lalu disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.17.



Gambar 3.17 Sequence Diagram fitur Sign-up the user to a system.

c. Sign-out the user of a system

Pada fitur ini dilakukan autentikasi user untuk dapat keluar dari dalam sistem. User akan *request sign out* yang divalidasi terlebih dahulu terhadap model User untuk dicek apakah masukan dari user sama dengan data yang disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.18.

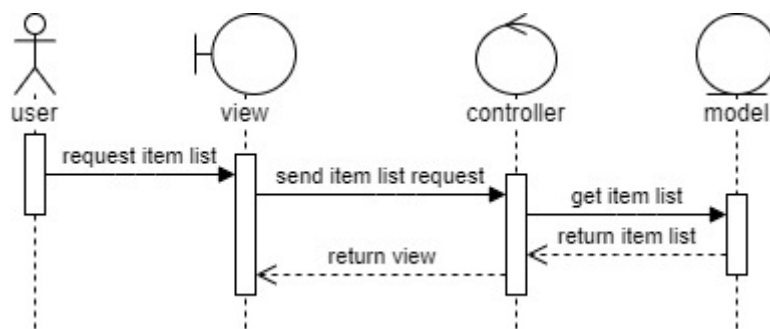


Gambar 3.18 *Sequence Diagram* fitur *Sign-out the user of a system*.

Bidding an item

a. Show the list of an item

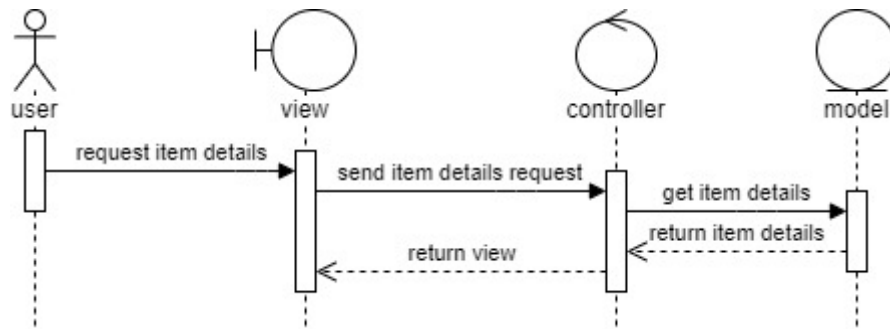
Pada fitur ini user akan meminta daftar item dengan mengirimkan *request* dahulu ke controller, kemudian user akan mendapatkan daftar item yang diminta, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.19.



Gambar 3.19 *Sequence Diagram* fitur *Show the list of an item*.

b. Show the details of an item

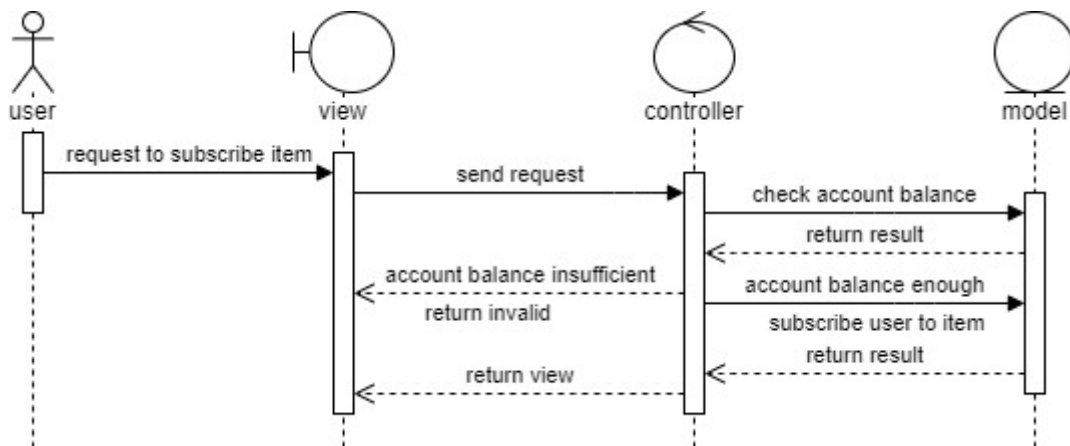
Pada fitur ini user akan meminta detail dari sebuah item dengan mengirimkan *request* dahulu ke controller, kemudian user akan mendapatkan detail dari item yang diminta, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.20.



Gambar 3.20 *Sequence Diagram* fitur *Show the details of an item*.

c. *Subscribe the current user to an item*

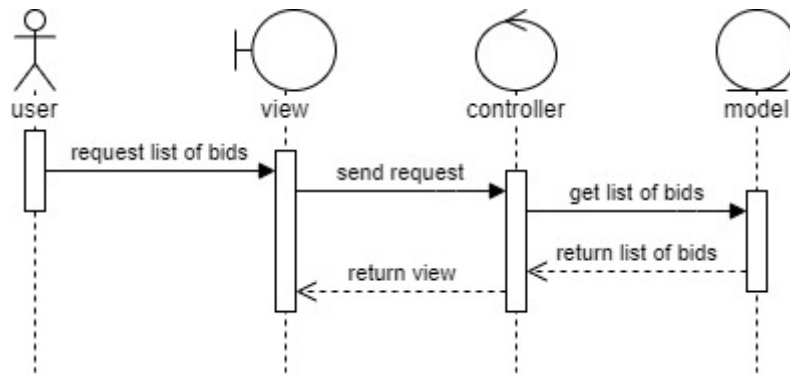
Pada fitur ini user akan meminta untuk daftar lelang, kemudian dilakukan pengecekan kecukupan saldo, yang apabila kurang akan *return invalid* dan bila mencukupi maka user akan terdaftar dalam lelang, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.21.



Gambar 3.21 *Sequence Diagram* fitur *Subscribe the current user to an item*.

d. *Show the bids of an item*

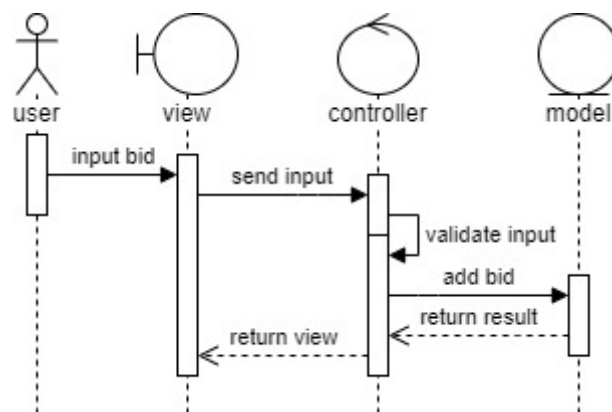
Pada fitur ini user akan meminta daftar lelang dari item dengan mengirimkan *request* dahulu ke controller, kemudian user akan mendapatkan daftar lelang dari item yang diminta, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.22.



Gambar 3.22 *Sequence Diagram* fitur *Show the bids of an item*.

e. Add the bid by a current user

Pada fitur ini user memasukkan input lelang dengan mengirimkannya ke controller yang juga akan melakukan validasi input. Kemudian, lelang akan dimasukkan setelah validasi, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.23.

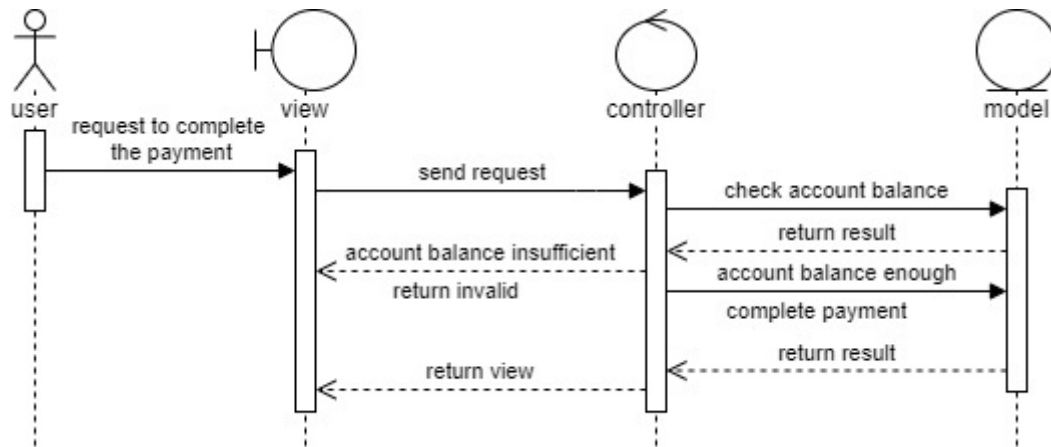


Gambar 3.23 *Sequence Diagram* fitur *Add the bid by a current user*.

Getting an item

a. Complete the payment of an auction

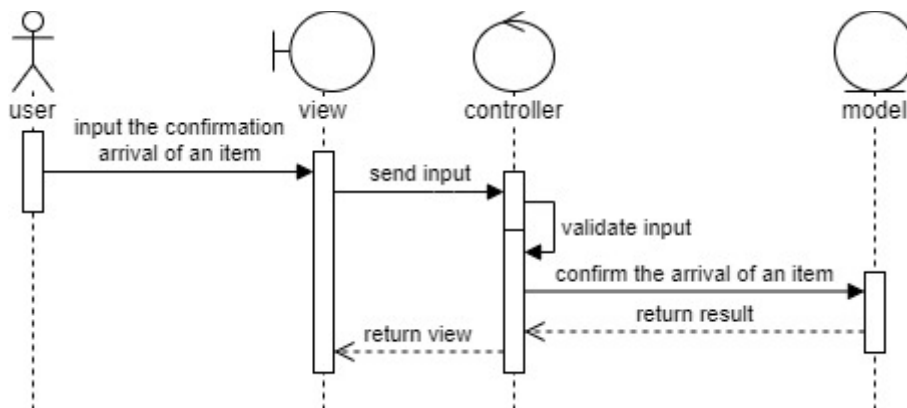
Pada fitur ini user yang akan menyelesaikan pembayaran melakukan *request* ke controller untuk dilakukan pengecekan kecukupan saldo, yang apabila kurang akan *return invalid* dan bila mencukupi maka user akan terselesaikan pembayarannya, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.24.



Gambar 3.24 *Sequence Diagram* fitur *Complete the payment of an auction*.

b. Confirm the arrival of an item

Pada fitur ini user akan diminta untuk input atau konfirmasi kedatangan item, yang kemudian dilakukan validasi input di controller, untuk konfirmasi kedatangan item tersebut setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.25.

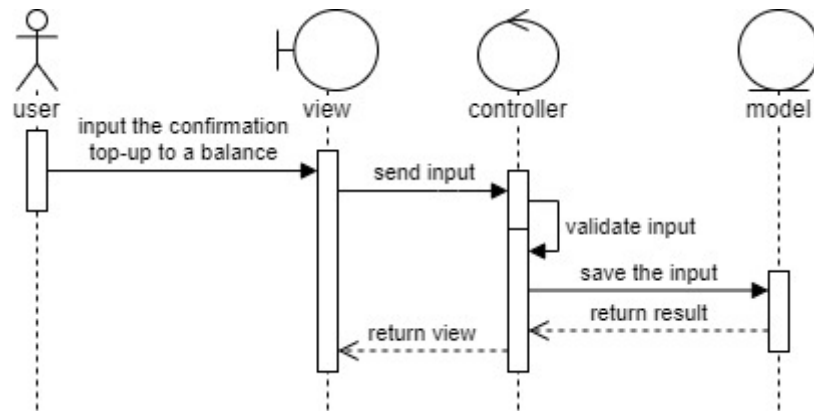


Gambar 3.25 *Sequence Diagram* fitur *Confirm the arrival of an item*.

Managing a balance

a. Confirm the top-up to a balance

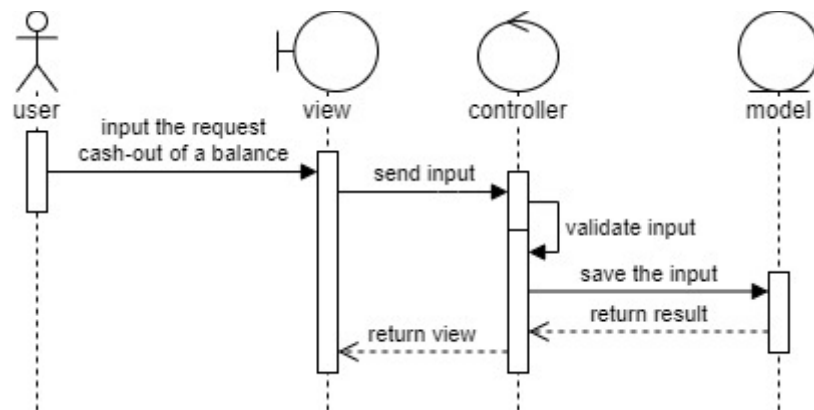
Pada fitur ini user akan diminta untuk memasukkan konfirmasi setelah top up saldo yang akan dikirimkan masuknya ke controller, kemudian dilakukan validasi input di controller, dan input akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.26.



Gambar 3.26 Sequence Diagram fitur *Confirm the top-up to a balance*.

b. Request the cash-out of a balance

Pada fitur ini user akan diminta untuk memasukkan *request* untuk menguangkan saldo yang akan dikirimkan masuknya ke controller, kemudian dilakukan validasi input di controller, dan input akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.27.

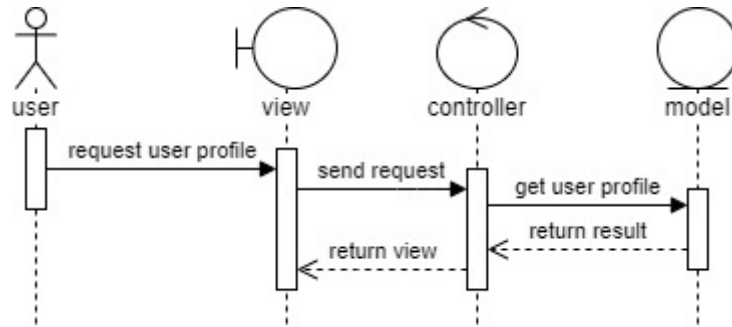


Gambar 3.27 Sequence Diagram fitur *Request the cash-out of a balance*.

Managing a profile

a. Show the profile of a current user

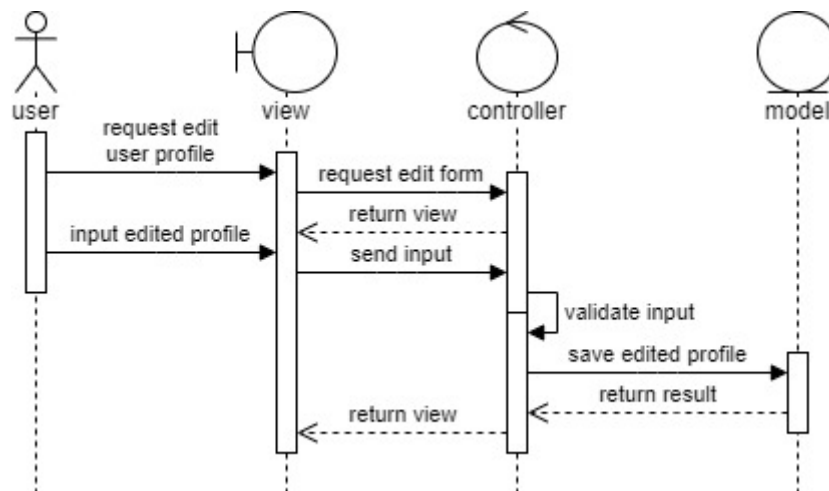
Pada fitur ini user akan meminta untuk melihat profil ke controller dan controller akan memperlihatkan profil user, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.28.



Gambar 3.28 *Sequence Diagram* fitur *Show the profile of a current user*.

b. Edit the profile of a current user

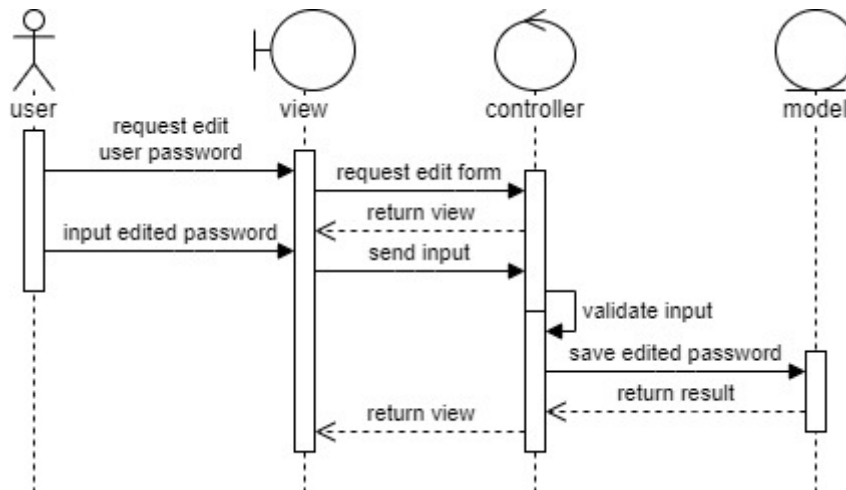
Pada fitur ini user yang ingin mengedit profil mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan profil yang diedit dan dikirimkan masuknya ke controller. Kemudian dilakukan validasi input di controller, dan input profil yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.29.



Gambar 3.29 *Sequence Diagram* fitur *Edit the profile of a current user*.

c. Edit the password of a current user

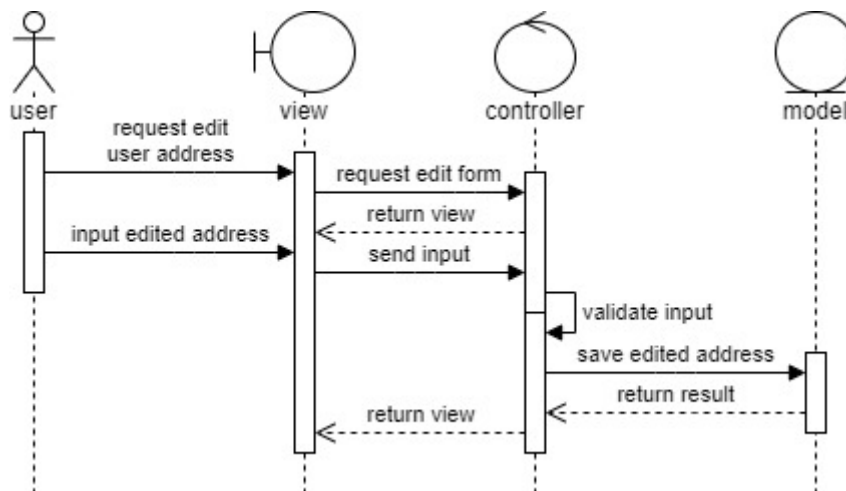
Pada fitur ini user yang ingin mengedit password mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan password yang diedit dan dikirimkan masuknya ke controller. Kemudian dilakukan validasi input di controller, dan input password yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.30.



Gambar 3.30 Sequence Diagram fitur *Edit the password of a current user*.

d. *Edit the address of a current user*

Pada fitur ini user yang ingin mengedit alamat mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan alamat yang diedit dan dikirimkan masukkannya ke controller. Kemudian dilakukan validasi input di controller, dan input alamat yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.31.

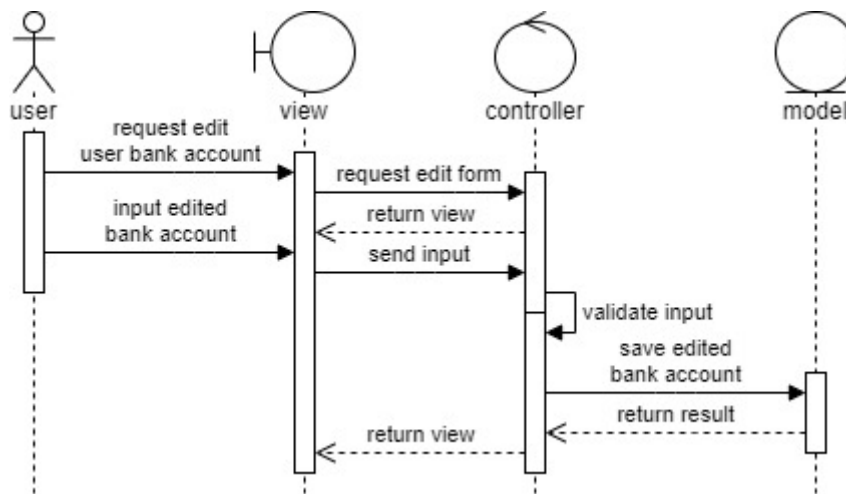


Gambar 3.31 Sequence Diagram fitur *Edit the address of a current user*.

e. *Edit the bank account of a current user*

Pada fitur ini user yang ingin mengedit data akun bank untuk penguangan saldo mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan data akun bank yang diedit dan dikirimkan masukkannya ke controller. Kemudian dilakukan validasi input di controller, dan input data akun bank yang sudah teredit akan

disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada user. Dimodelkan seperti pada Gambar 3.32.



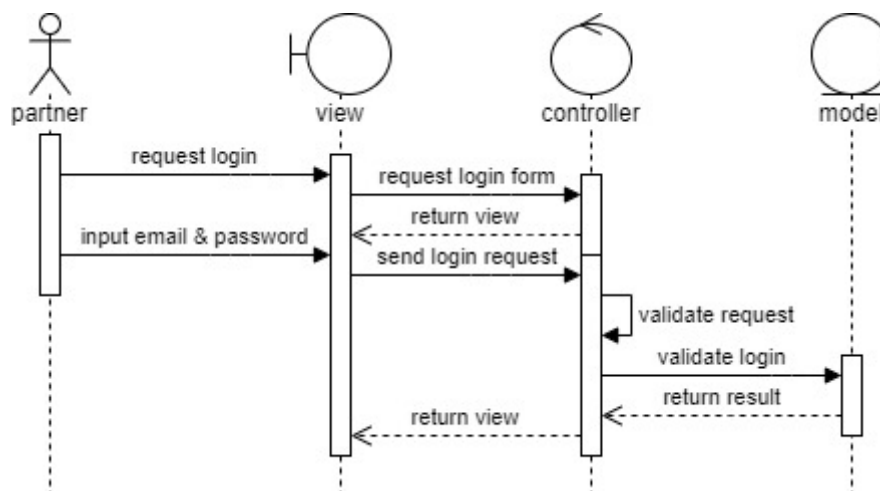
Gambar 3.32 *Sequence Diagram* fitur *Edit the bank account of a current user*.

B. Aktor Partner

Authenticating a partner

a. *Sign-in the partner to a system*

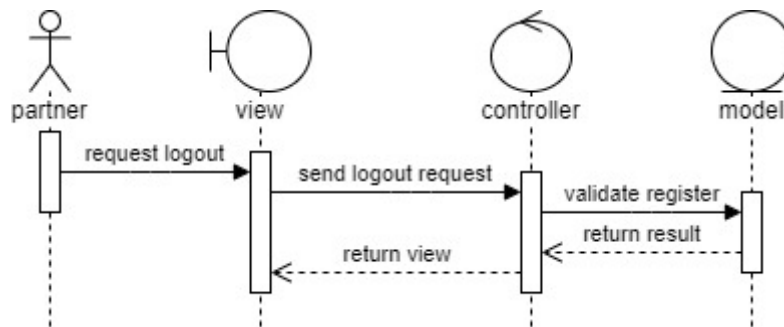
Pada fitur ini dilakukan autentikasi partner untuk dapat masuk ke dalam sistem. Partner yang sudah memasukkan *email* dan *password*-nya kemudian divalidasi terlebih dahulu masukkannya di *controller*, lalu divalidasi lagi terhadap model Partner untuk dicek apakah masukan dari partner sama dengan data yang disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.33.



Gambar 3.33 *Sequence Diagram* fitur *Sign-in the partner to a system*.

b. *Sign-out the partner of a system*

Pada fitur ini dilakukan autentikasi partner untuk dapat keluar dari sistem dengan *request logout* dan divalidasi *registernya*, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.34.

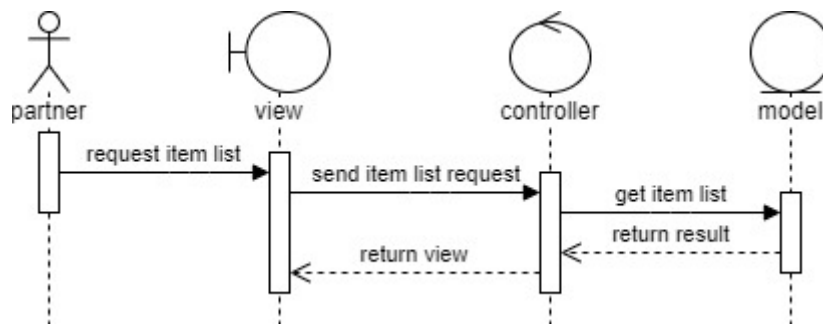


Gambar 3.34 *Sequence Diagram* fitur *Sign-out the partner of a system*.

Managing an item

a. *Show the item list by a current partner*

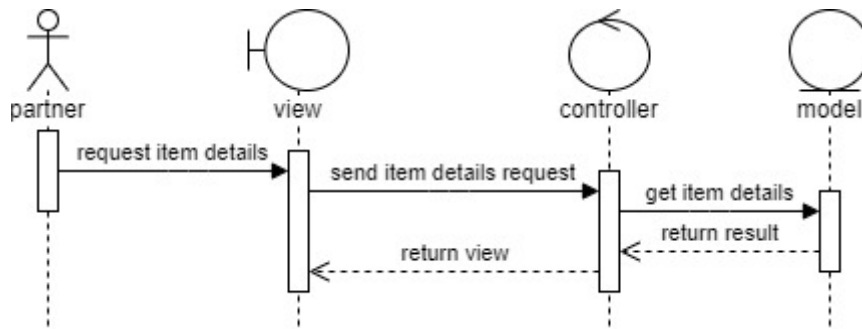
Pada fitur ini partner yang akan meminta daftar item dengan mengirimkan *request* dahulu ke controller, kemudian user akan mendapatkan daftar item yang diminta, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.35.



Gambar 3.35 *Sequence Diagram* fitur *Show the item list by a current partner*.

b. *Show the details of an item*

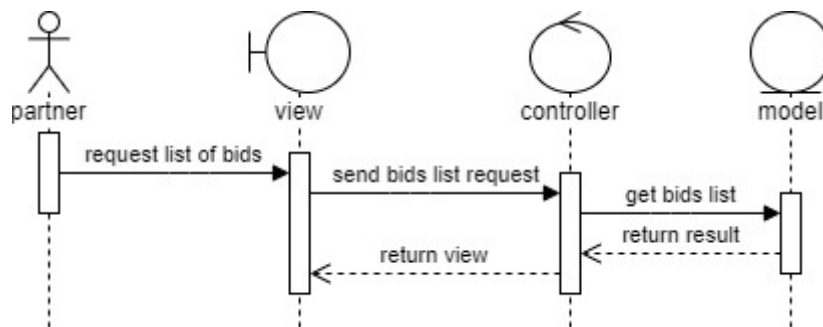
Pada fitur ini partner akan meminta detail dari sebuah item dengan mengirimkan *request* dahulu ke controller, kemudian partner akan mendapatkan detail dari item yang diminta, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.36.



Gambar 3.36 *Sequence Diagram* fitur *Show the details of an item*.

c. *Show the bids of an item*

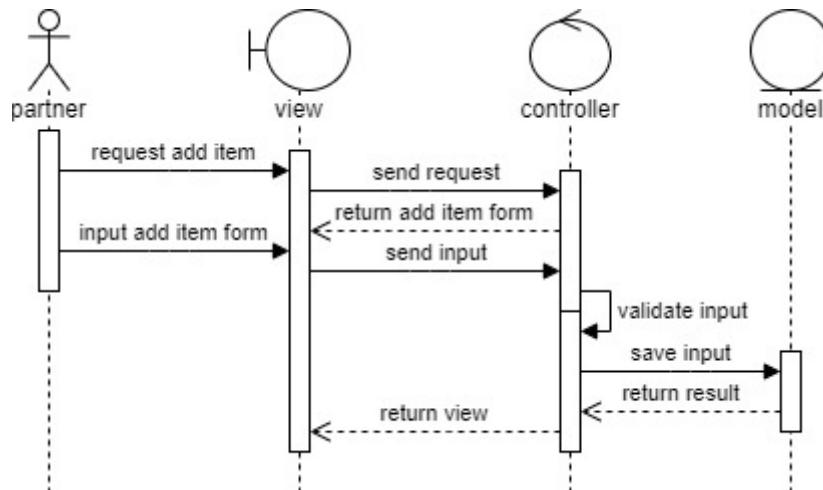
Pada fitur ini partner akan meminta daftar lelang dari item dengan mengirimkan *request* dahulu ke controller, kemudian partner akan mendapatkan daftar lelang dari item yang diminta, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.37.



Gambar 3.37 *Sequence Diagram* fitur *Show the bids of an item*.

d. *Add the item by a current partner*

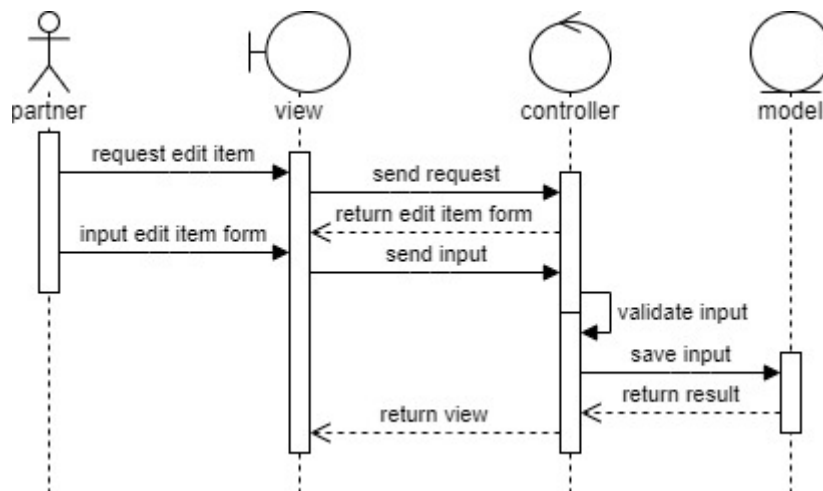
Pada fitur ini partner menambahkan item dengan mengirimkan *request* dengan formulir dan mengisi serta memasukkannya ke sistem. Partner yang sudah memasukkan formulir tadi kemudian divalidasi terlebih dahulu masukkannya di *controller*, lalu masukkannya akan disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.38.



Gambar 3.38 *Sequence Diagram* fitur *Add the item by a current partner*.

e. *Edit the item by a current partner*

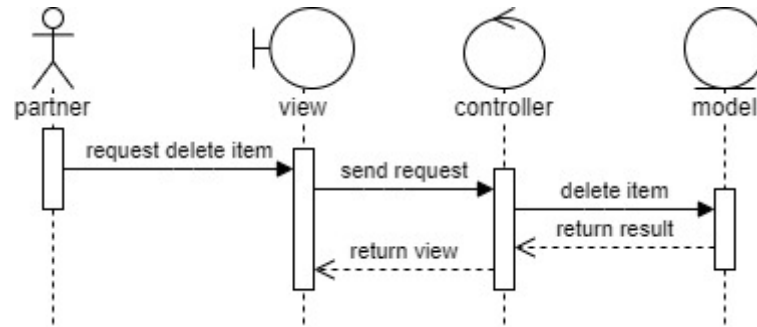
Pada fitur ini partner mengedit item dengan mengirimkan *request* dengan formulir dan mengisi serta memasukkannya ke sistem. Partner yang sudah memasukkan formulir tadi kemudian divalidasi terlebih dahulu masukkannya di *controller*, lalu masukkannya akan disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.39.



Gambar 3.39 *Sequence Diagram* fitur *Edit the item by a current partner*.

f. *Delete the item by a current partner*

Pada fitur ini partner dapat menghapus item dengan mengirimkan *request* untuk penghapusan item. Item akan dihapus dan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.40.

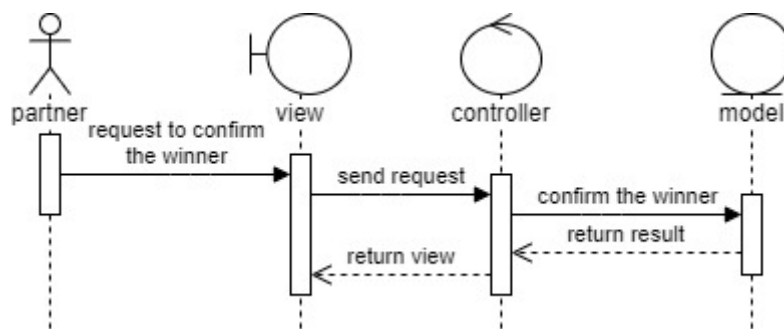


Gambar 3.40 *Sequence Diagram* fitur *Delete the item by a current partner*.

Shipping an item

a. Confirm the winner of an auction

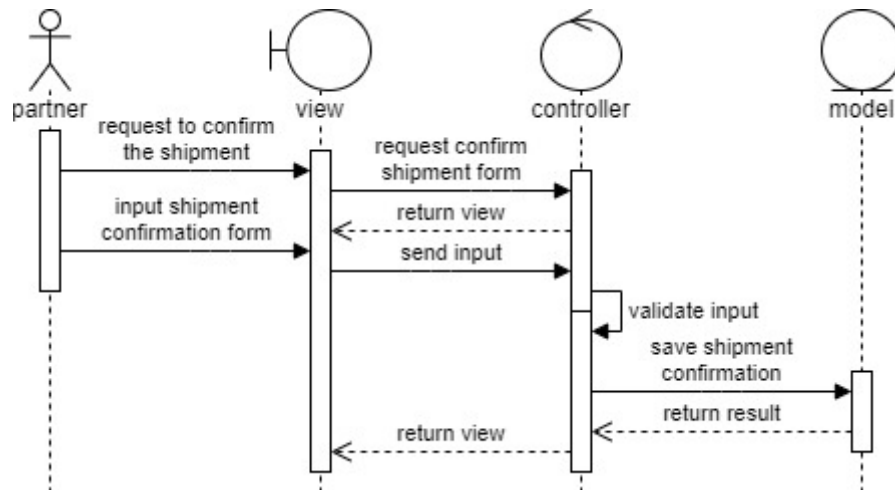
Pada fitur ini partner *request* untuk mengkonfirmasi pemenang lelang dengan mengirimkan *request* tersebut. Pemenang lelang akan terkonfirmasi dan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.41.



Gambar 3.41 *Sequence Diagram* fitur *Confirm the winner of an auction*.

b. Confirm the shipment of an item

Pada fitur ini partner mengkonfirmasi pengiriman dengan mengirimkan *request* dengan formulir dan mengisi serta memasukkannya ke sistem. Partner yang sudah memasukkan formulir tadi kemudian divalidasi terlebih dahulu masukkannya di *controller*, lalu konfirmasi pengiriman akan disimpan dalam database, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.42.

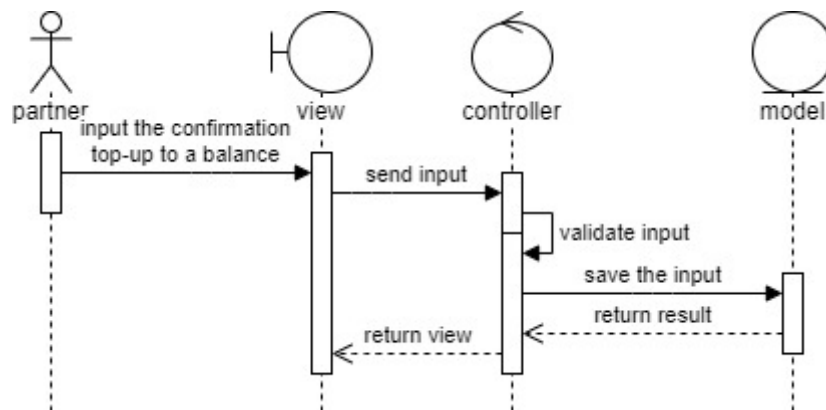


Gambar 3.42 Sequence Diagram fitur *Confirm the shipment of an item*.

Managing a balance

a. Confirm the top-up for a balance

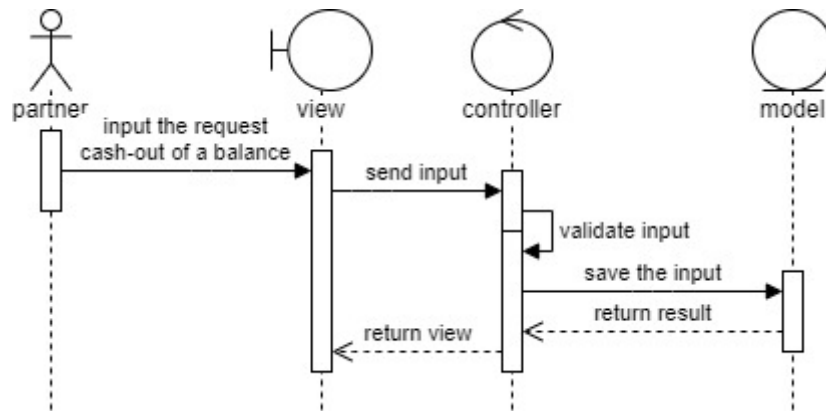
Pada fitur ini partner akan diminta untuk memasukkan konfirmasi setelah top up saldo yang akan dikirimkan masuknya ke controller, kemudian dilakukan validasi input di controller, dan input akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.43.



Gambar 3.43 Sequence Diagram fitur *Confirm the top-up for a balance*.

b. Request the cash-out of a balance

Pada fitur ini partner akan diminta untuk memasukkan *request* untuk menguangkan saldo yang akan dikirimkan masuknya ke controller, kemudian dilakukan validasi input di controller, dan input akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.44.

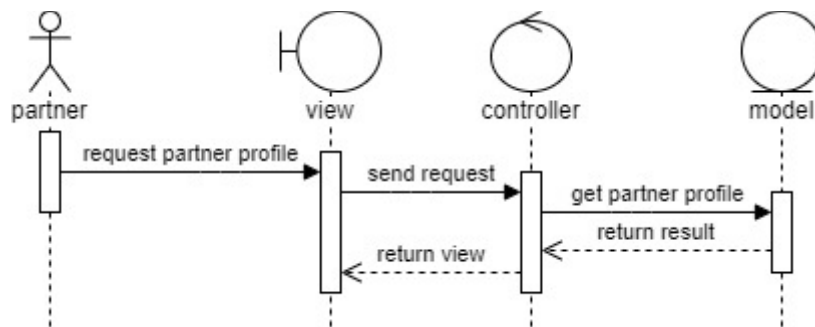


Gambar 3.44 *Sequence Diagram* fitur *Request the cash-out of a balance*.

Managing a profile

a. Show the profile of a current partner

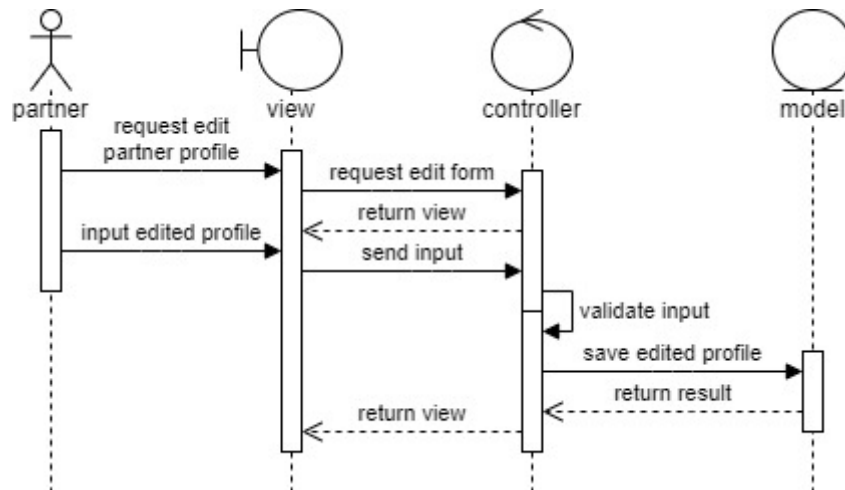
Pada fitur ini partner akan meminta untuk melihat profil ke controller dan controller akan memperlihatkan profil partner, setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.45.



Gambar 3.45 *Sequence Diagram* fitur *Show the profile of a current partner*.

b. Edit the profile of a current partner

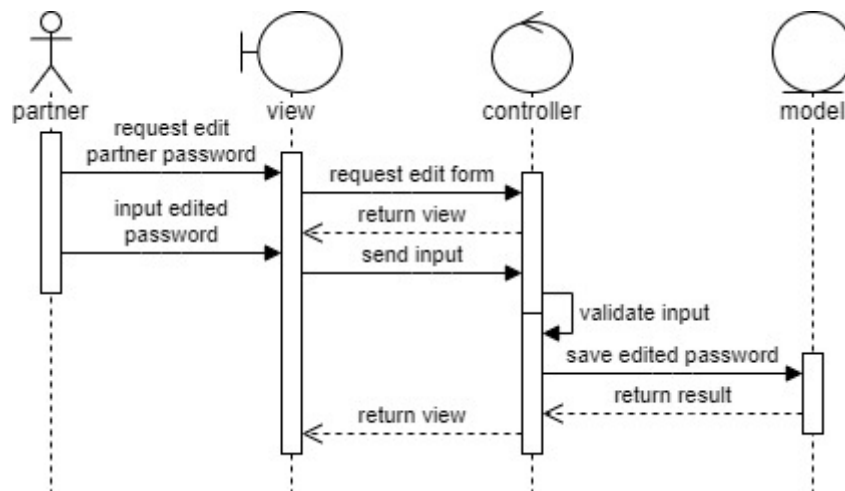
Pada fitur ini partner yang ingin mengedit profil mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan profil yang diedit dan dikirimkan masuknya ke controller. Kemudian dilakukan validasi input di controller, dan input profil yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.46.



Gambar 3.46 Sequence Diagram fitur *Edit the profile of a current partner*.

c. *Edit the password of a current partner*

Pada fitur ini partner yang ingin mengedit password mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan password yang diedit dan dikirimkan masuknya ke controller. Kemudian dilakukan validasi input di controller, dan input password yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.47.

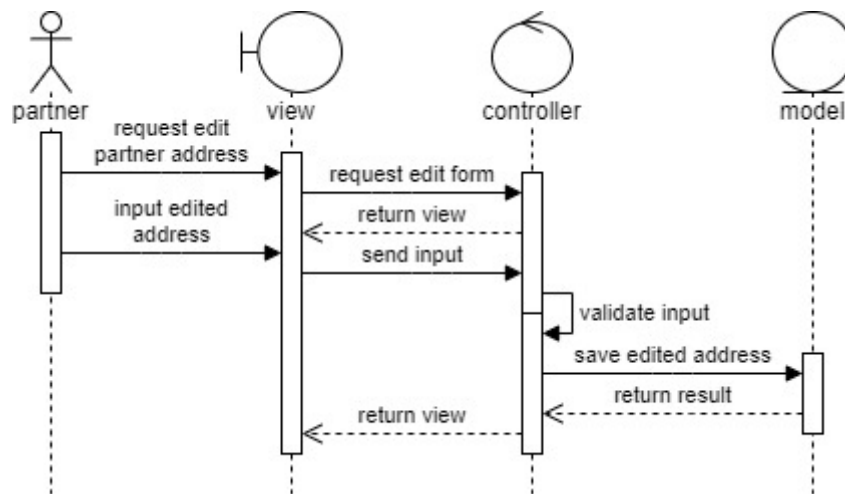


Gambar 3.47 Sequence Diagram fitur *Edit the password of a current partner*.

d. *Edit the address of a current partner*

Pada fitur ini partner yang ingin mengedit alamat mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan alamat yang diedit dan dikirimkan masuknya ke controller. Kemudian dilakukan validasi input di controller, dan input alamat

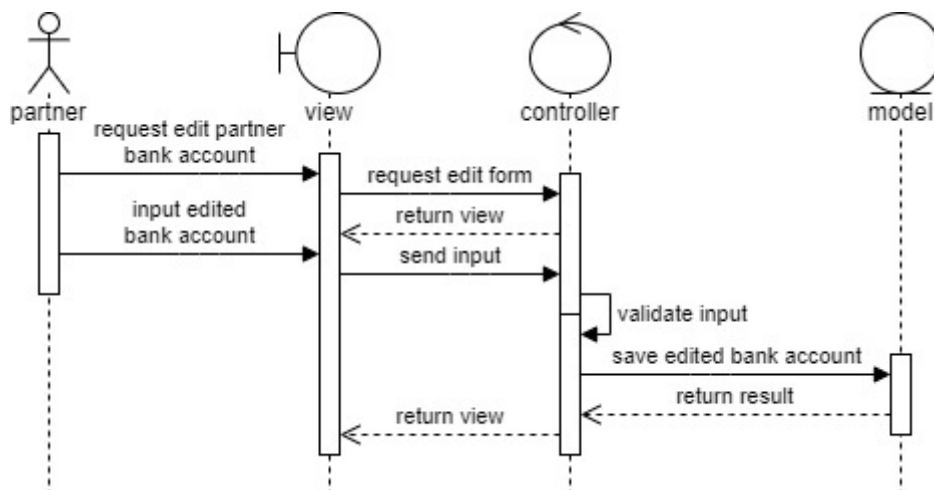
yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.48.



Gambar 3.48 Sequence Diagram fitur *Edit the address of a current partner*.

e. *Edit the bank account of a current partner*

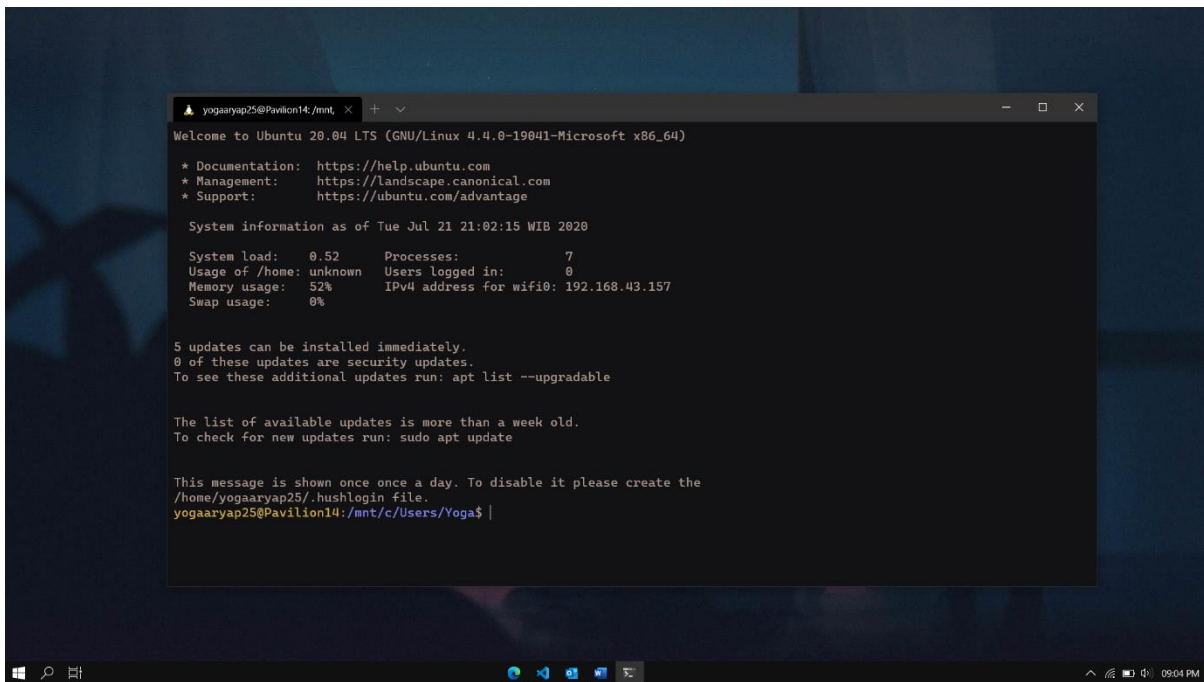
Pada fitur ini partner yang ingin mengedit data akun bank untuk penguangan saldo mengirimkan *request* untuk formulir pengeditan dan kembali ke *return view* untuk memasukkan data akun bank yang diedit dan dikirimkan masukkannya ke controller. Kemudian dilakukan validasi input di controller, dan input data akun bank yang sudah teredit akan disimpan setelah itu baru di-*return* hasilnya dan pada akhirnya di-*return view* kepada partner. Dimodelkan seperti pada Gambar 3.49.



Gambar 3.49 Sequence Diagram fitur *Edit the bank account of a current partner*.

3.5 Build by Feature

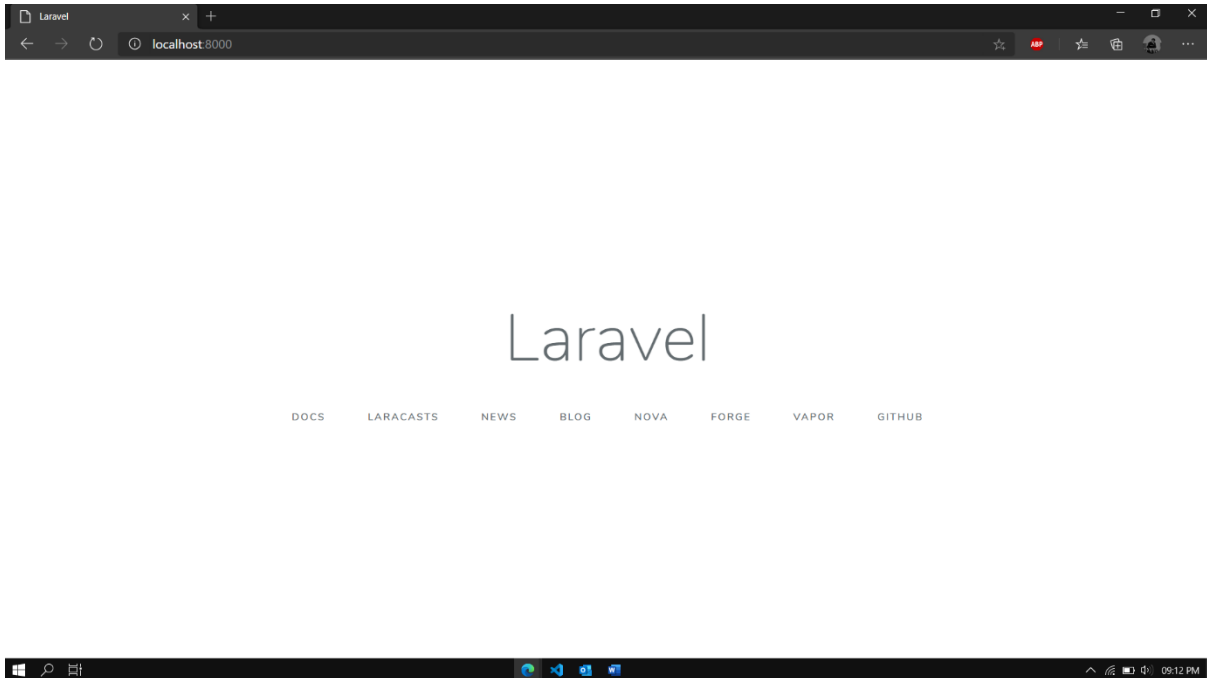
Fase terakhir pada metode pengembangan *Feature-driven Development*, *Build by Feature*, kami membangun sistem fitur demi fitur yang sudah dideklarasikan pada fase *Build a Features List* dan kurang lebih sesuai rancangan pada fase *Design by Feature*. Pembangunan sistem dilakukan secara lokal terlebih dahulu, menggunakan Windows Subsystem for Linux (WSL) dengan *distro* Ubuntu 20.04 LTS seperti pada Gambar 3.50.



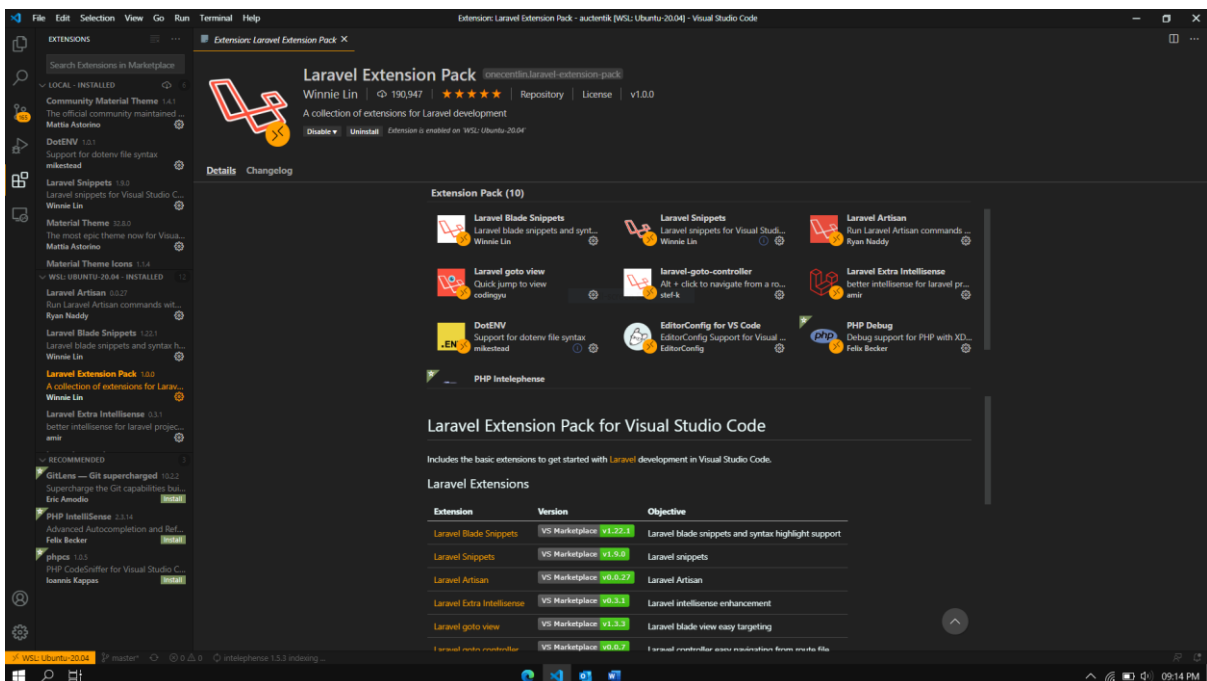
Gambar 3.50 Tampilan Windows Subsystem for Linux Ubuntu 20.04.

Digunakannya WSL pada pengembangan ini agar dapat lebih mudah menggunakan fitur-fitur yang sudah ada secara bawaan pada Ubuntu daripada harus menginstall dan menkonfigurasi terlebih dahulu apabila pada Windows, seperti git, node, dan npm. *Server* lokal juga dijalankan pada WSL Ubuntu dengan *service* apache2, untuk database menggunakan *service* mysql, dan juga menggunakan phpMyAdmin untuk dapat mempermudah olah data pada database. Untuk mempermudah pembangunan, dikarenakan sistem yang akan dibangun adalah berbasis *web*, maka digunakan *framework* Laravel yang sudah tergolong lengkap untuk pengembangan *web*. Untuk *front-end* dengan menggunakan *framework* Laravel sudah ada bawaan seperti Bootstrap dan jQuery, juga menggunakan beberapa *plugin-plugin* tambahan untuk dapat mempermudah. Instalasi Laravel menggunakan PHP Dependency Manager Composer, untuk versi Laravel yang digunakan adalah versi 6 dan menghasilkan tampilan seperti pada Gambar 3.51 setelah dilakukan instalasi dan menjalankan *build server* untuk

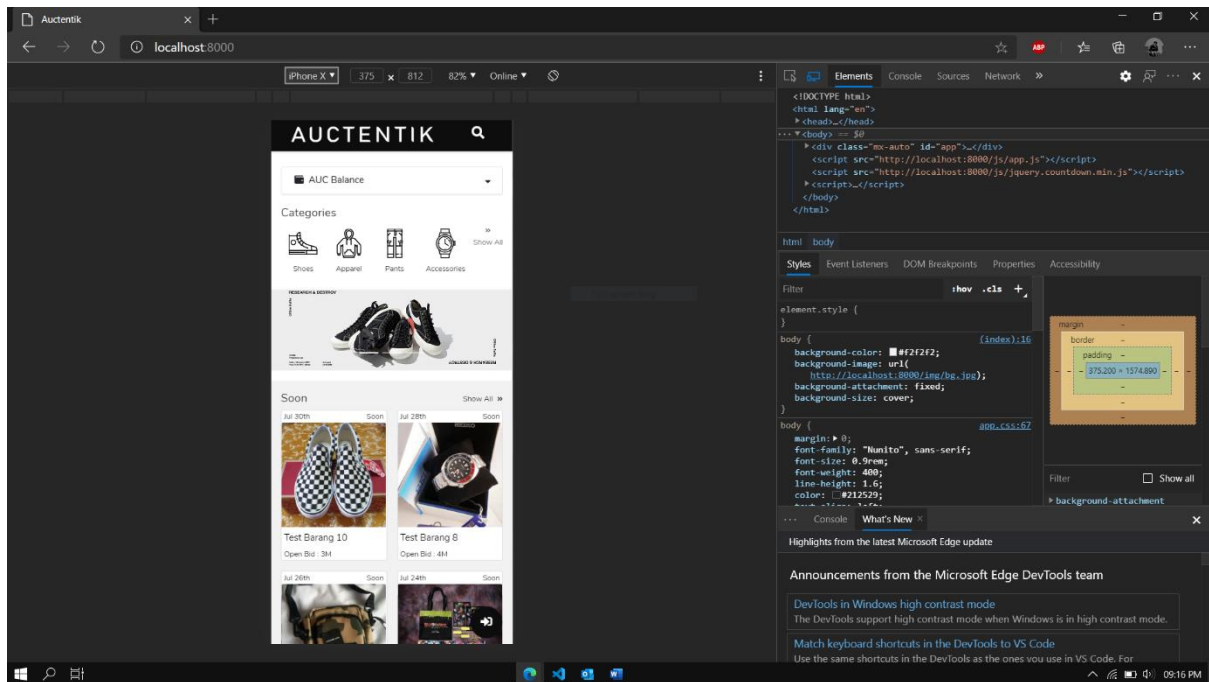
pertama kali-nya. Untuk *text editor* dan *debugging*-nya menggunakan Microsoft Visual Studio Code dan Microsoft Edge, dikarenakan adanya dukungan *plugin-plugin* seperti pada Gambar 3.52 yang dapat mempermudah dalam *edit code* dan *debugging* seperti pada Gambar 3.53.



Gambar 3.51 Tampilan Laravel setelah instalasi.

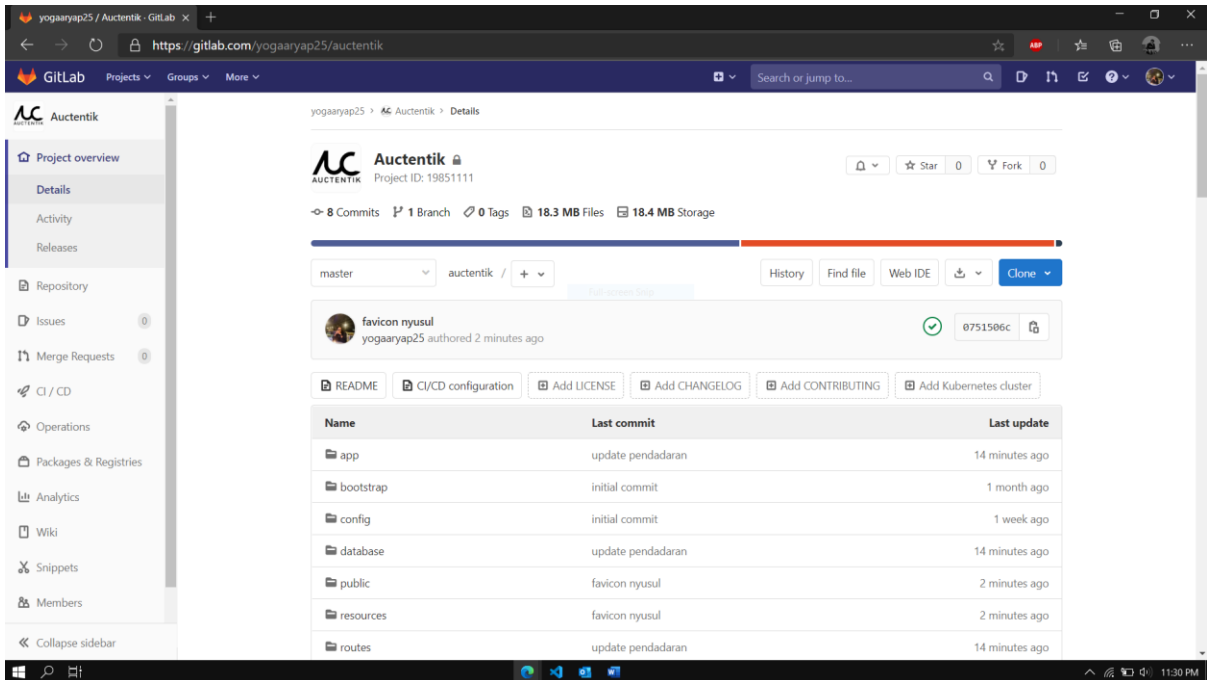


Gambar 3.52 Salah satu *plugin* yang digunakan pada Microsoft Visual Studio Code.

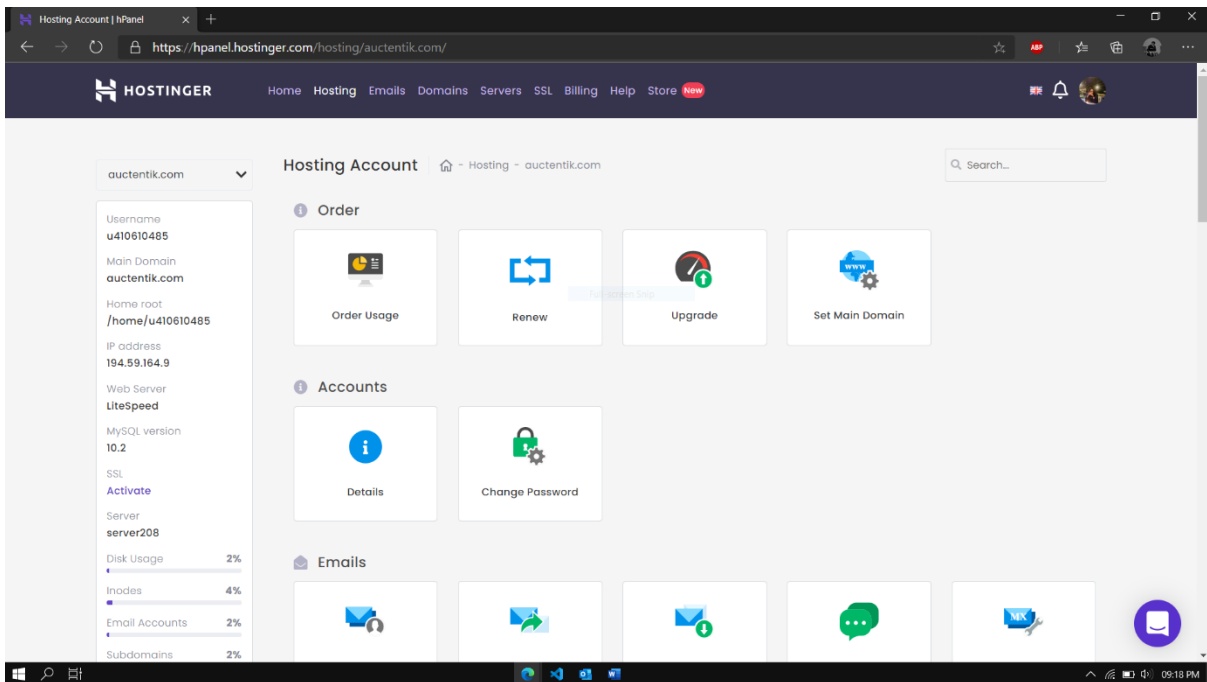


Gambar 3.53 *Debugging* menggunakan Microsoft Edge.

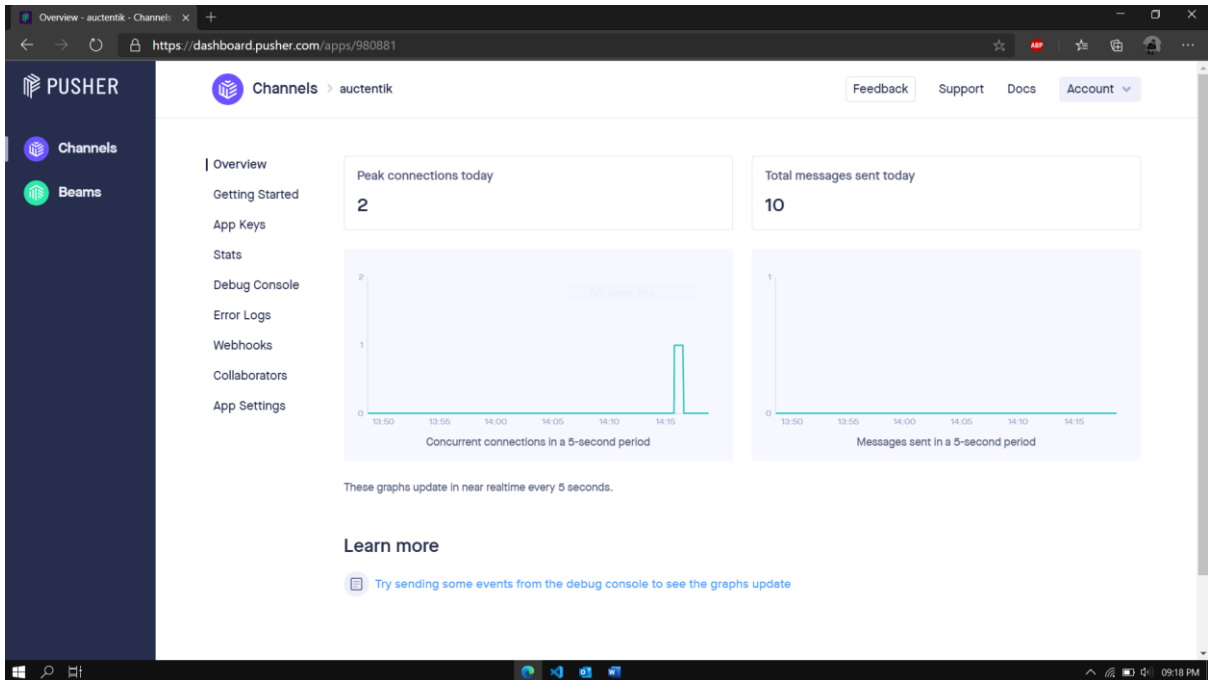
Setelah installasi Laravel berhasil, kami melakukan konfigurasi pada git. Kami menggunakan layanan GitLab untuk menyimpan kode sistem Auctentik, seperti pada Gambar 3.54, dikarenakan pada GitLab *project* dapat diset *private* secara gratis. Setelah *project* berhasil tersimpan di GitLab, kami melakukan konfigurasi untuk CI/CD-nya (*Continuous Integration/Continuous Delivery*). Digunakannya CI/CD pada pengembangan ini adalah agar setiap fitur yang selesai dibangun dapat langsung diakses oleh klien. Untuk hostingnya sendiri kami menggunakan *provider* Hostinger (Gambar 3.55). Kami juga menggunakan *hosted service* Pusher (Gambar 3.56) untuk *real-time* datanya yang digunakan pada proses lelang agar tawaran dari pengguna dapat ter-*update* secara langsung. Setelah itu mulai dibangun fitur demi fitur yang sudah dideklarasikan pada fase *Build a Features List* dan kurang lebih sesuai rancangan pada fase *Design by Feature*.



Gambar 3.54 Tampilan GitLab.



Gambar 3.55 Tampilan Hostinger



Gambar 3.56 Tampilan Pusher.

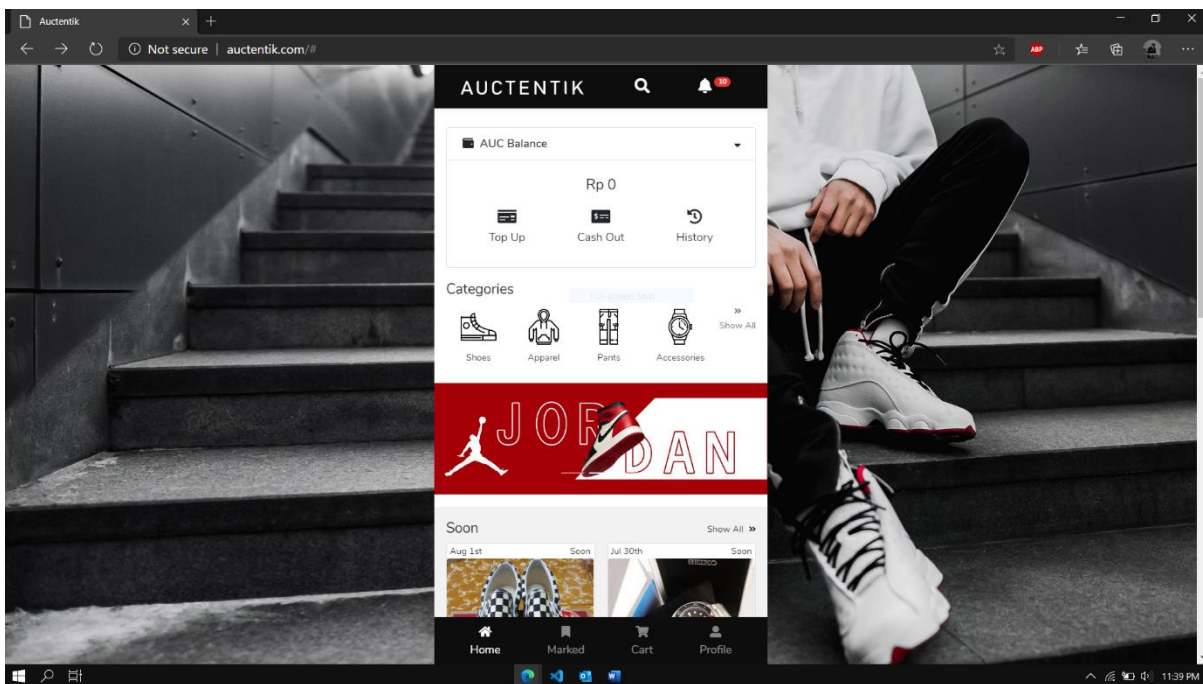
BAB IV

HASIL DAN PEMBAHASAN

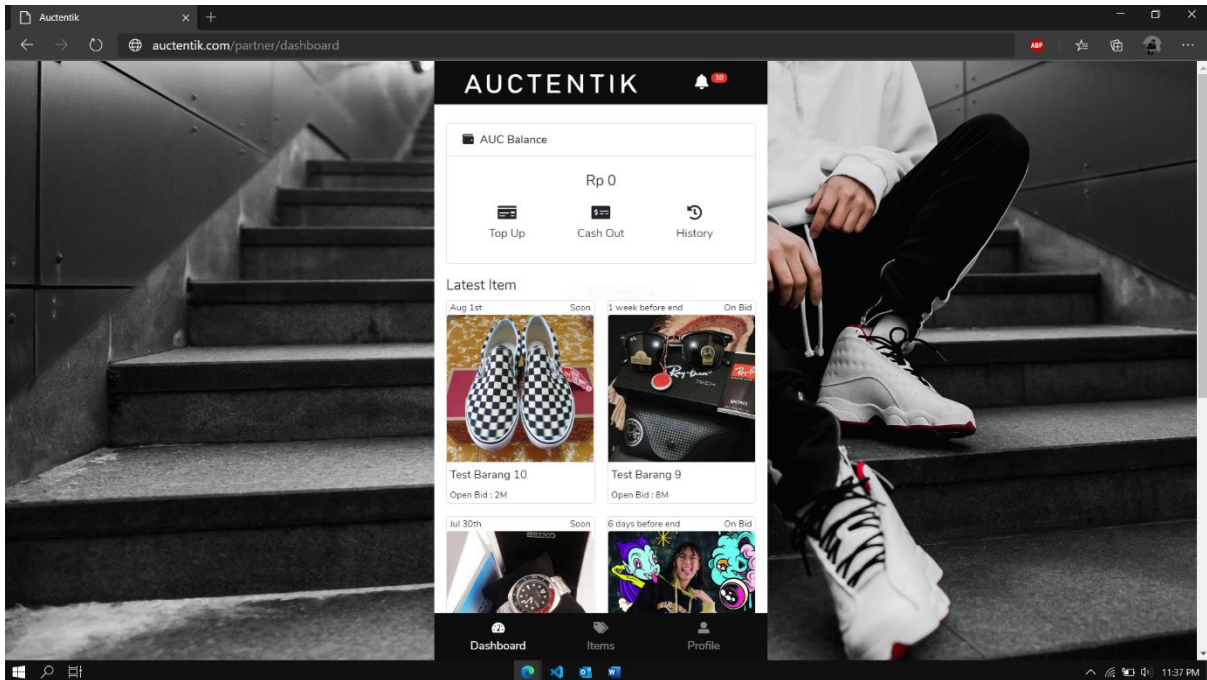
4.1 Hasil

Berdasarkan hasil pengembangan sistem Auctentik yang berbasis web menggunakan *framework* Laravel dengan metode *Feature-driven Development* (FDD) yang sudah dilakukan, selanjutnya dilakukan *hosting* dengan *domain* pada alamat <http://auctentik.com/>.

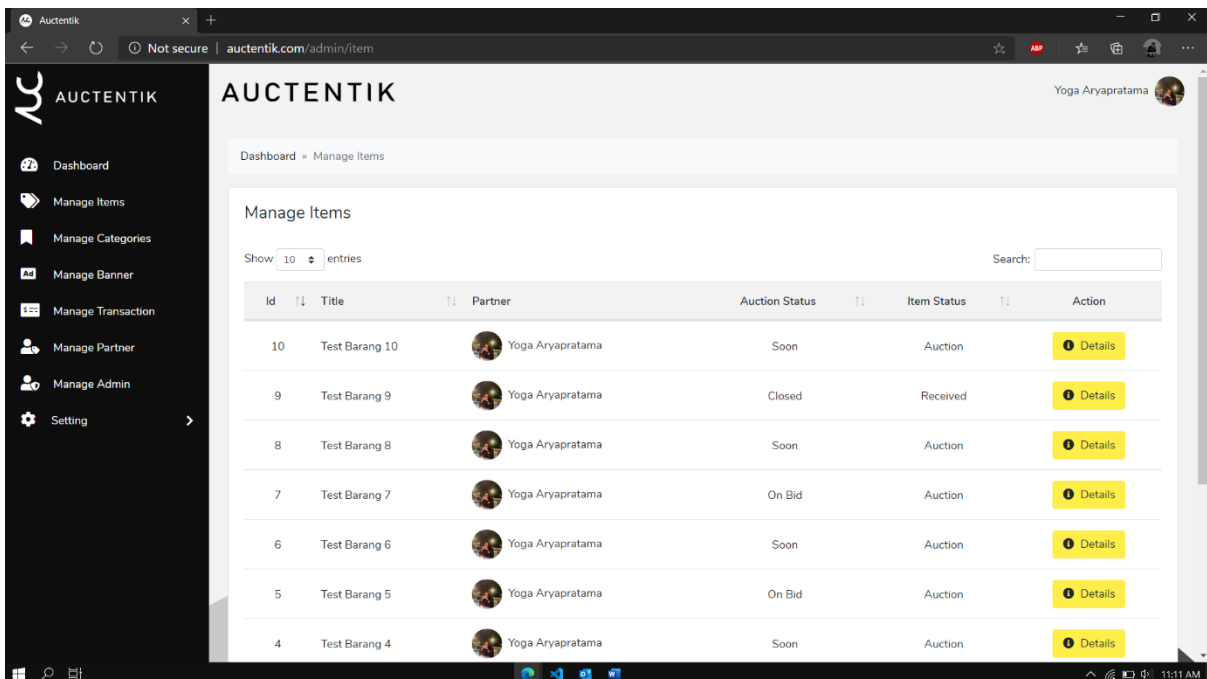
Secara tampilan, dari sisi pengguna dan partner, dikembangkan dengan fokus penggunaan *mobile*. Sehingga meskipun diakses dengan menggunakan *desktop*, tampilan akan tetap menggunakan *width mobile* yaitu 425px seperti pada Gambar 4.1 dan Gambar 4.2. Sedangkan dari sisi admin, tampilan dikembangkan dengan fokus penggunaan *desktop* seperti pada Gambar 4.3.



Gambar 4.1 Tampilan pengguna pada *desktop*.



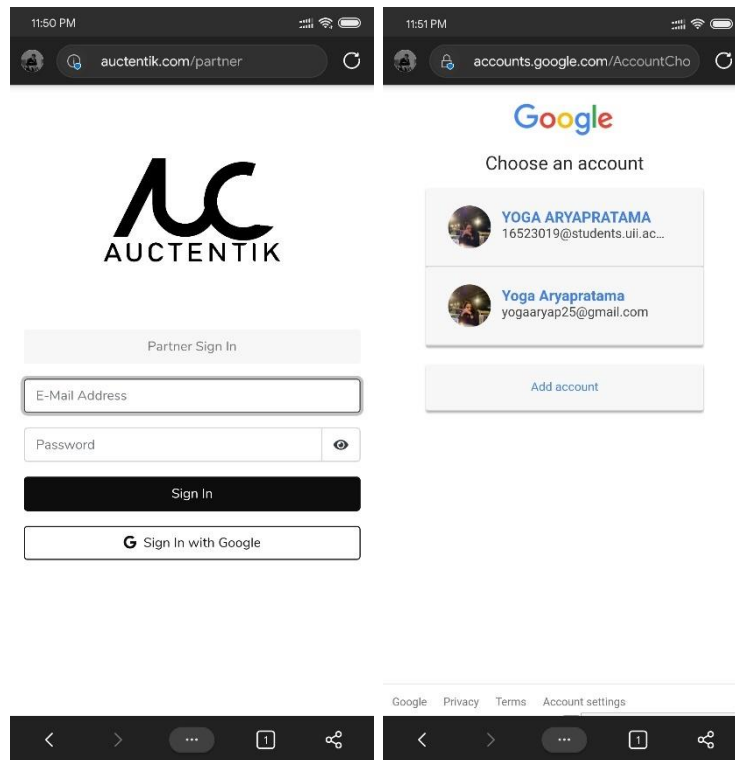
Gambar 4.2 Tampilan partner pada *desktop*.



Gambar 4.3 Tampilan admin pada *desktop*.

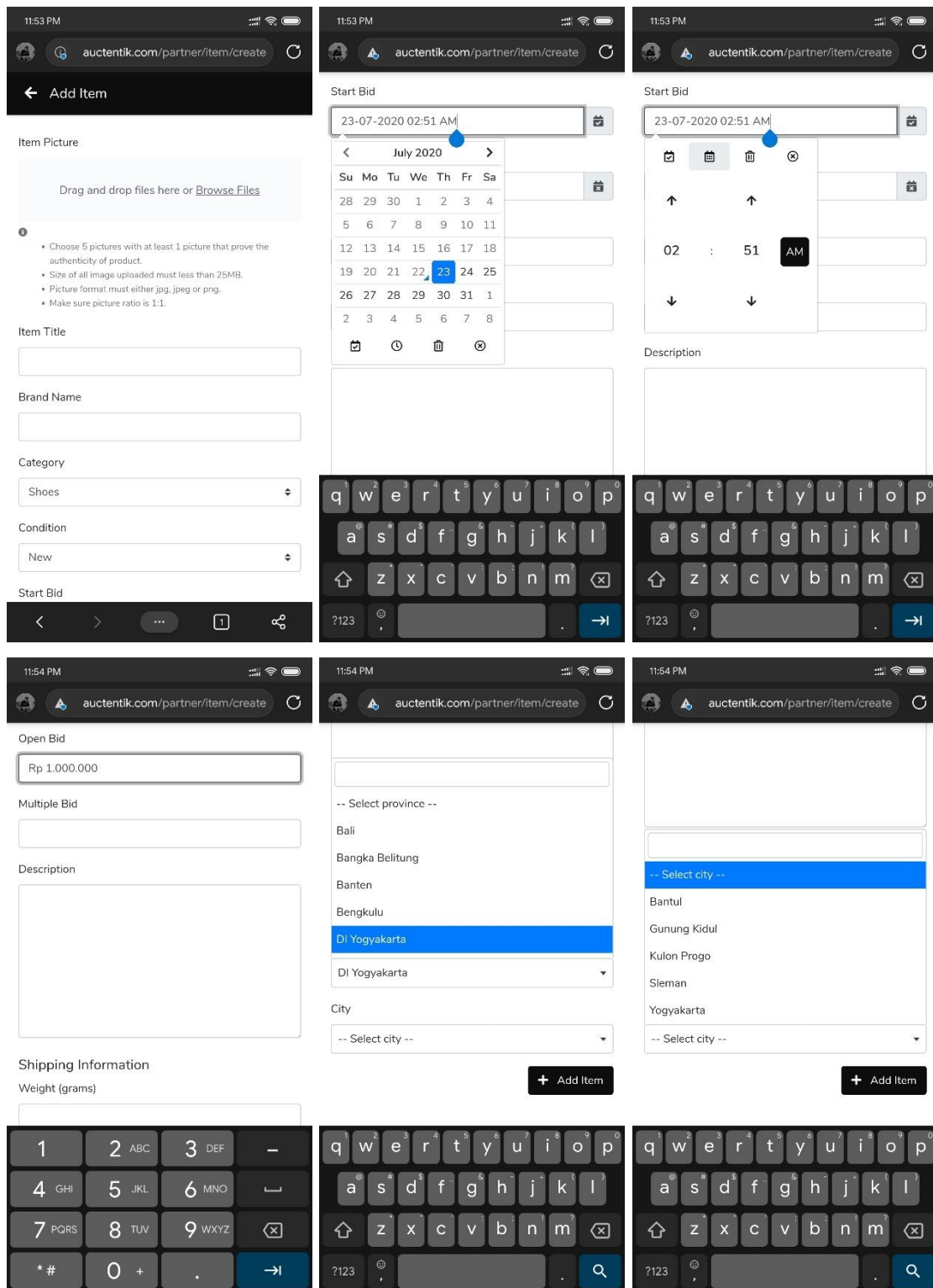
Secara penggunaan, sesuai dengan proses bisnis sistem (Gambar 3.3 & Gambar 3.4) yang sudah didefinisikan pada fase *Develop an Overall Model*, dijabarkan sebagai berikut:

a. Partner masuk ke dalam sistem (Gambar 4.4).



Gambar 4.4 Autentikasi Partner

b. Partner menambahkan barang yang akan dilelang (Gambar 4.5).



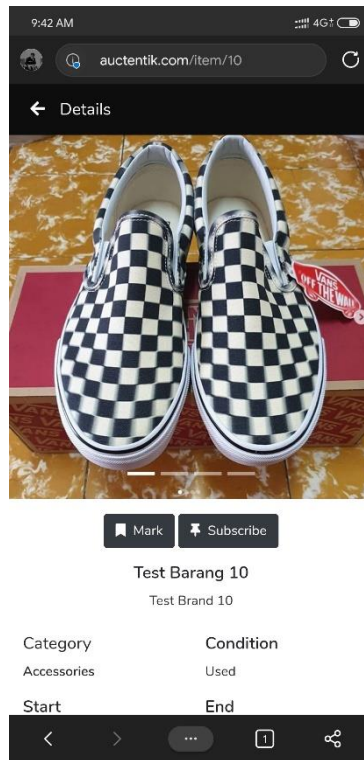
Gambar 4.5 Menambahkan barang yang akan dilelang.

c. Pengguna masuk ke dalam sistem (Gambar 4.6).

The image displays two side-by-side screenshots of the Auctentik mobile application interface. Both screens show the time as 11:55 PM and the URL as auctentik.com/login and auctentik.com/register respectively. The left screen is the login page, featuring the Auctentik logo, an 'E-Mail Address' input field, a 'Password' input field with a visibility toggle, a 'Sign In' button, a 'Register' button, and a 'Sign Up with Google' button. The right screen is the registration page, featuring the Auctentik logo, a 'Name' input field, an 'E-Mail Address' input field, a 'Password' input field, a 'Confirm Password' input field, a 'Register' button, a 'Sign In' button, and a 'Sign Up with Google' button. Both screens have a dark navigation bar at the bottom with back, forward, and search icons.

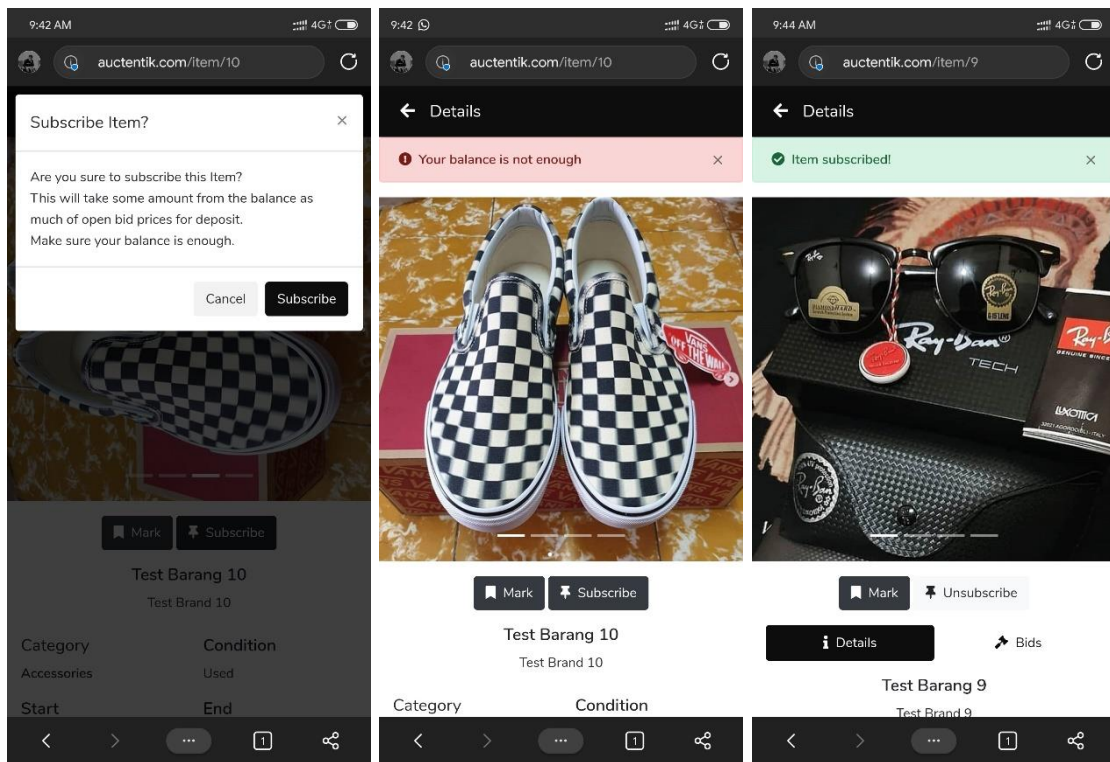
Gambar 4.6 Autentikasi pengguna.

d. Pengguna melihat barang yang akan dilelang atau sedang dilelang (Gambar 4.7).



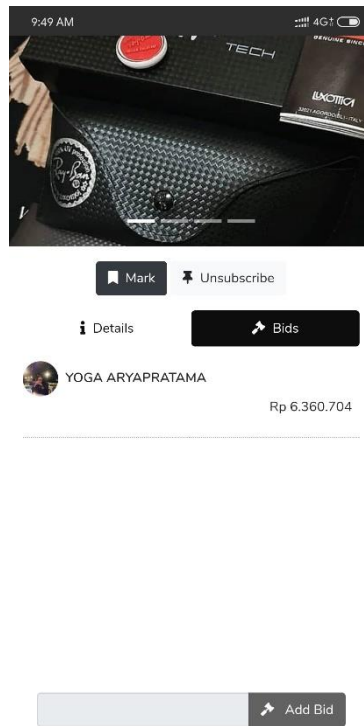
Gambar 4.7 Barang yang dilelang.

e. Pengguna mendaftar lelang (Gambar 4.8).



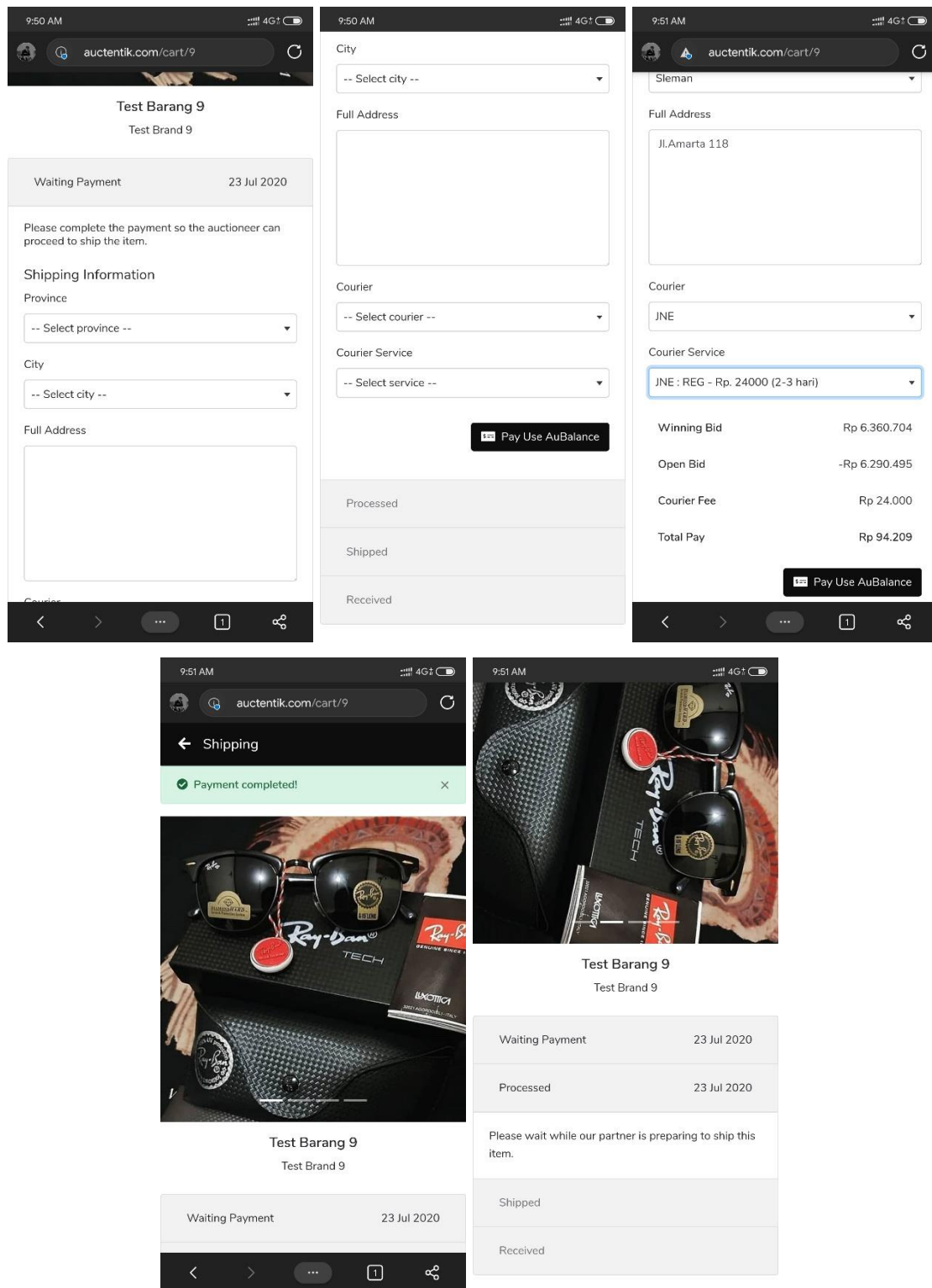
Gambar 4.8 Mendaftar lelang.

f. Pengguna mengikuti lelang (Gambar 4.9).



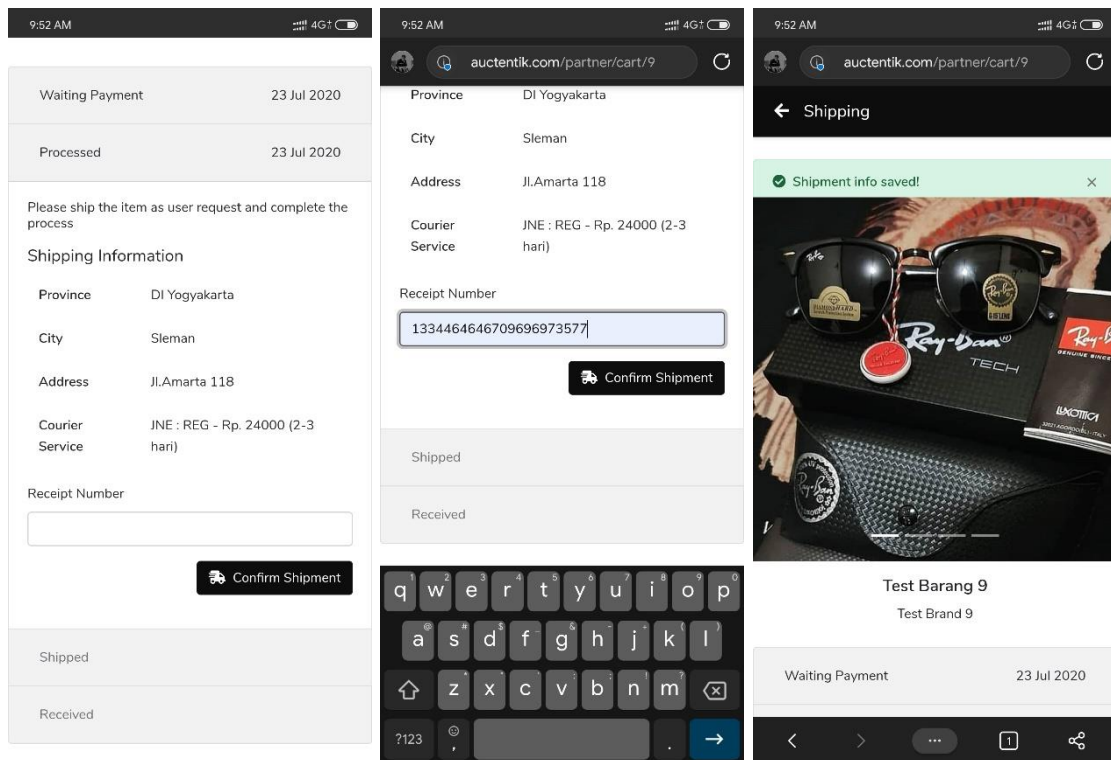
Gambar 4.9 Mengikuti lelang.

g. Pengguna melakukan pelunasan lelang (Gambar 4.10).



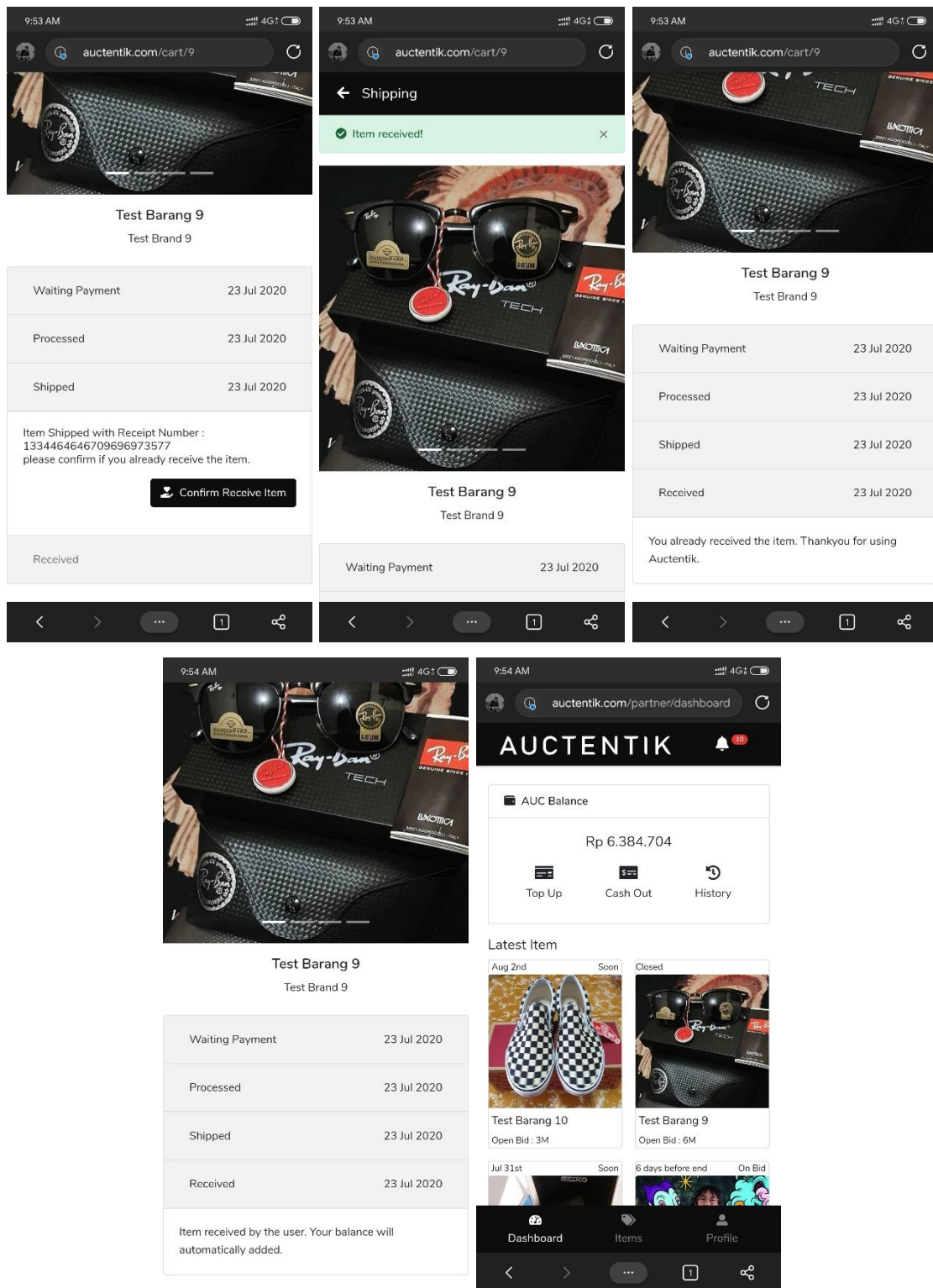
Gambar 4.10 Pelunasan apabila memenangkan lelang.

h. Partner mengirimkan barang (Gambar 4.11).



Gambar 4.11 Mengirim barang yang dilelang.

i. Pengguna melakukan konfirmasi penerimaan barang (Gambar 4.12).



Gambar 4.12 Konfirmasi penerimaan barang.

4.2 Pembahasan

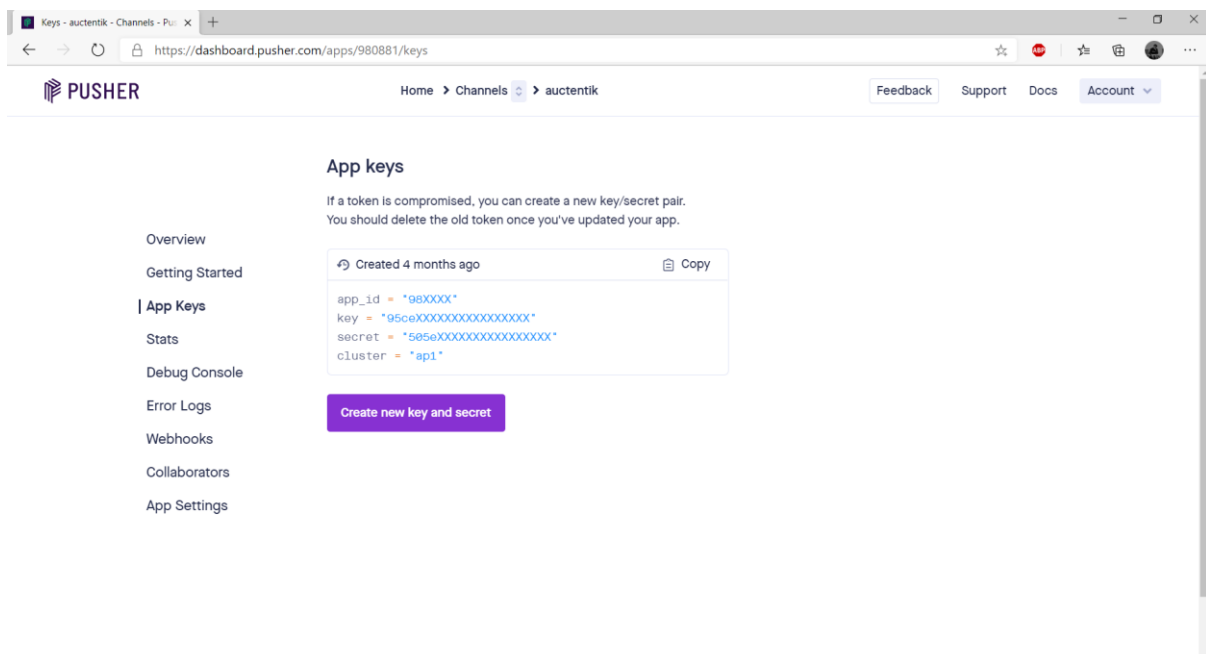
Berdasarkan hasil pengembangan sistem Aucantik yang berbasis web menggunakan *framework* Laravel dengan metode *Feature-driven Development* (FDD) yang sudah dilakukan, didapatkan beberapa hal yang dapat dibahas seperti penyelesaian masalah, pengembangan adaptif dan pengembangan cepat.

4.2.1 Penyelesaian Masalah

Seperti yang sudah didefinisikan pada fase *Develop an Overall Model*, ada tiga permasalahan utama yang ditemukan pada sistem lelang *online* yang ada saat ini, yaitu *jump bid*, *bid & run*, dan *doubt of authenticity*, dari ketiga masalah yang hadir kami mencoba menawarkan solusi dengan sistem yang dikembangkan ini.

Jump bid

Jump bid adalah keadaan dimana penawar melakukan penawaran terhadap barang yang dilelang dengan nominal yang melebihi ketentuan awal dari pelemang. Kami menawarkan pengembangan solusinya yang berupa *bid by system*. Tahap pertama pada pengembangan fitur *bid by system*, kami mendaftar dan membuat *channel project* baru pada *hosted service* Pusher, dan didapat *key* untuk *channel* tersebut seperti pada Gambar 4.13. *Key* yang sudah didapat lalu dimasukkan pada file `.env` seperti pada Gambar 4.14.



Gambar 4.13 *App keys* yang didapat setelah membuat *channel* pada Pusher

```

...
BROADCAST_DRIVER=pusher

PUSHER_APP_ID=980XXX
PUSHER_APP_KEY=95ceXXXXXXXXXXXXXXXXXX
PUSHER_APP_SECRET=505eXXXXXXXXXXXXXXXXXX
PUSHER_APP_CLUSTER=ap1
...

```

Gambar 4.14 Baris kode yang ditambahkan pada file `.env`

Langkah berikutnya kami membuat file `Bids.vue` dan `Form.vue` di dalam *folder* `resources/js/components`. File `Bids.vue` digunakan untuk menampilkan penawaran-penawaran yang diajukan oleh pengguna seperti ditunjukkan pada Gambar 4.15, sedangkan file `Form.vue` digunakan untuk menampilkan nominal penawaran berikutnya dan *button* untuk mengajukan penawaran seperti ditunjukkan pada Gambar 4.16.

```

<template>
  <ul class="bid">
    <li class="left clearfix" v-for="(bid, index) in bids" :key="bid + '-' +
index">
      <div class="bid-body clearfix">
        <div class="header">
          <strong class="primary-font">
            {{ bid.user.name }}
          </strong>
        </div>
        <p class="float-right mr-2">
          {{ 'Rp ' + bid.bid.toLocaleString('id') }}
        </p>
      </div>
    </li>
  </ul>
</template>

<script>
export default {
  props: ['bids'],
}
</script>

```

Gambar 4.15 Baris kode pada file `Bids.vue`

```

<template>
  <div class="input-group mb-3">
    <input type="text" name="bid" id="newBid" class="form-control" readonly v-
model="bid">
    <div class="input-group-append">
      <button class="btn btn-primary" id="btn-bid" @click="submit" v-
bind:disabled="bid == ''">
        <i class="fas fa-gavel mr-2"></i>
        Add Bid
      </button>
    </div>
  </div>
</template>

<script>
export default {
  props: ['user', 'item', 'bids'],
  methods: {
    submit() {
      this.$emit('sent', {
        user: this.user,
        item: this.item,
        bid: this.bid.replace(/^[^0-9]+/g, '')
      });
    },
  },
  computed: {
    bid () {
      if (this.bids.length) {
        if (this.bids[this.bids.length - 1].user.id == this.user.id) {
          return '';
        } else {
          return 'Rp ' + (parseInt(this.bids[this.bids.length - 1].bid) +
item.multiple).toLocaleString('id');
        }
      }
      return 'Rp ' + (item.ob + item.multiple).toLocaleString('id');
    }
  }
}
</script>

```

Gambar 4.16 Baris kode pada file Form.vue

Kemudian kami membuat file `bid.js` di dalam *folder* `resources/js`, file ini digunakan untuk mendeklarasikan dua file sebelumnya sebagai komponen Vue dan untuk *men-handle request* atau *input* dari pengguna perihal penawaran pada lelang seperti ditunjukkan pada Gambar 4.17.

```

require('./bootstrap');
import Vue from 'vue';
import Bids from './components/Bids.vue';
import Form from './components/Form.vue';
Vue.component('bids', Bids);
Vue.component('bid-form', Form);

const app = new Vue({
  el: '#app',
  data: {
    bids: []
  },
  created() {
    this.fetchBids();

    Echo.private('item.'+item.id)
      .listen('BidAdded', (e) => {
        this.bids.push({
          bid: e.bid.bid,
          user: e.user
        })
      })
  },
  methods: {
    fetchBids() {
      axios.get('/bids/'+item.id).then(response => {
        this.bids = response.data;
      });
    },
    addBid(bid) {
      this.bids.push(bid);
      axios.post('/bids/'+item.id, bid).then(response => {
        console.log(response.data);
      });
    }
  }
});

```

Gambar 4.17 Baris kode pada file `bid.js`

Selanjutnya komponen dipanggil pada *view* yaitu pada file `resources/views/user/item/show.blade.php` dengan menambahkan *custom tag* yakni `<bids></bids>` dan `<bid-form></bid-form>` juga memanggil *script* agar dapat ditampilkan pada tampilan barang yang dilelang seperti pada Gambar 4.18.

```

...
<div class="tab-pane fade" id="pills-bids" role="tabpanel" aria-labelledby="pills-
bids-tab">
  <div class="container-mt-3" ref="scrollParent" id="bidBox">
    <bids :bids="bids"></bids>
  </div>
  <div class="container mt-2">
    <bid-form v-on:sent="addBid" :bids="bids" :user="{ auth()-
>user() }" :item="{ $item }" ref="bidForm"></bid-form>
  </div>
</div>
@push('script')
  <script src="{ asset('js/bid.js') }"></script>
@endpush
...

```

Gambar 4.18 Baris kode pada file `show.blade.php`

Berikutnya kami membuat file `BidController.php` pada folder `app/Http/Controllers/User` yang digunakan untuk menyimpan fungsi `getBids()` dan `broadcastBid()`. Fungsi `getBids()` digunakan untuk mengambil penawaran-penawaran pada barang yang sedang dilelang, sedangkan fungsi `broadcastBid()` digunakan untuk melakukan *broadcast* dan menyimpan penawaran baru yang masuk dari pengguna, seperti ditunjukkan pada Gambar 4.19.

```

...
public function getBids(Item $item)
{
    return Bid::where('item_id', $item->id)->with('user', 'item')->get();
}

public function broadcastBid(Item $item, Request $request)
{
    $user = Auth::user();
    $bid = Bid::create([
        'user_id' => $user->id,
        'item_id' => $item->id,
        'bid' => (int) $request->bid
    ]);
    broadcast(new BidAdded($item, $user, $bid)->toOthers());
    return response()->json(['status' => 'Bid Added!']);
}
...

```

Gambar 4.19 Baris kode pada file `BidController.php`

Selanjutnya membuat *class event* `BidAdded`, dengan menjalankan *command* `php artisan make:event BidAdded` pada *command line*, setelah itu akan ter-*create* file `BidAdded.php` di dalam folder `app/Events` dan diisi seperti pada Gambar 4.20.

```

...
use Dispatchable, InteractsWithSockets, SerializesModels;
public $item;
public $user;
public $bid;

public function __construct(Item $item, User $user, Bid $bid)
{
    $this->item = $item;
    $this->user = $user;
    $this->bid = $bid;
}

public function broadcastOn()
{
    return new PrivateChannel('item.'.$this->item->id);
}
...

```

Gambar 4.20 Baris kode pada file `BidAdded.php`

Langkah terakhir menambahkan *routing* untuk *private channel* dan aplikasinya. Pada file `routes/channel.php` ditambah baris kode seperti pada Gambar 4.21, sedangkan pada file `routes/web.php` ditambah baris kode seperti pada Gambar 4.22.

```

...
Broadcast::channel('item.{item}', function ($user, $item) {
    return Auth::check();
});
...

```

Gambar 4.21 Baris kode pada file `routes/channel.php`

```

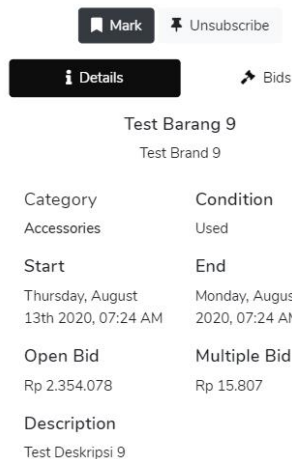
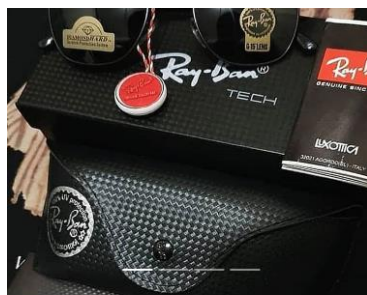
...
Route::get('/bids/{item}', 'User/BidController@getBids');
Route::post('/bids/{item}', 'User/BidController@broadcastBid');
...

```

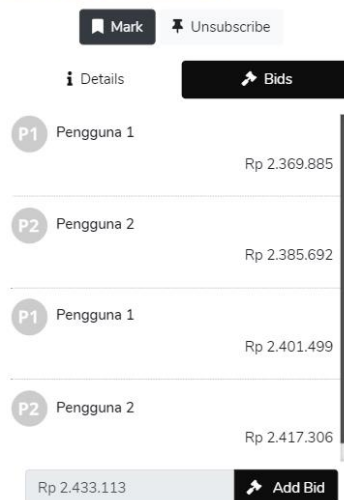
Gambar 4.22 Baris kode pada file `routes/web.php`

Implementasi pada sistem ketika dijalankan adalah pada proses lelang, nominal penawaran pertama akan sama dengan nominal *open bid* ditambah dengan nominal *multiple bid* yang ditunjukkan pada Gambar 4.23, begitu juga penawaran-penawaran selanjutnya akan sama nominalnya dengan penawaran sebelumnya ditambah dengan nominal *multiple bid*. Penawaran baru yang diajukan oleh salah satu pengguna akan otomatis muncul pada tampilan pengguna lainnya dan memperbarui *form* untuk mengajukan dengan nominal penawaran terakhir ditambah dengan nominal *multiple bid* seperti ditunjukkan pada Gambar 4.24. Pengguna dengan nominal penawaran tertinggi juga akan di-*disable form*-nya sehingga tidak dapat mengajukan penawaran secara berturut seperti ditunjukkan pada Gambar 4.25. Otomatisasi ini

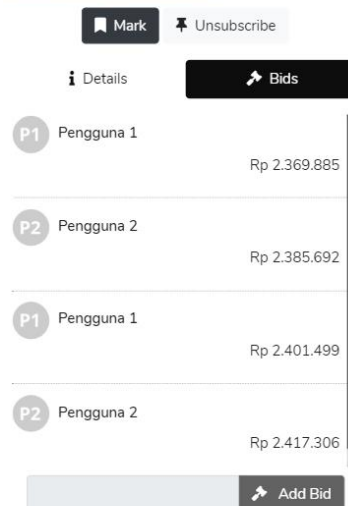
secara langsung akan membuat praktik *jump bid* tidak bisa dilakukan lagi, karena semua nominal penawaran sudah sesuai dengan aturan yang ditentukan oleh pelelang.



Gambar 4.23 Detail barang yang dilelang



Gambar 4.24 Tampilan lelang pengguna 1

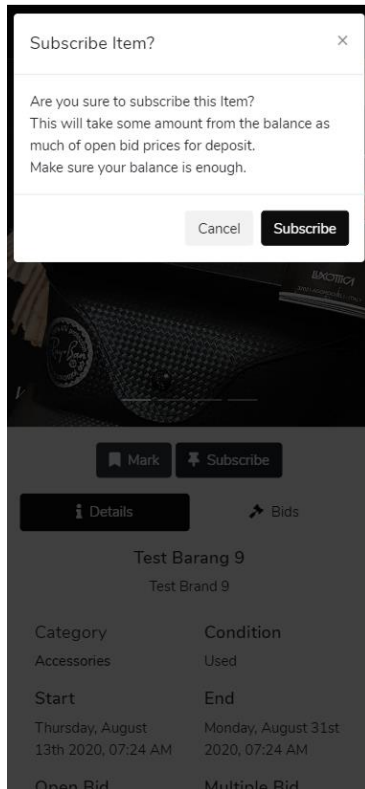


Gambar 4.25 Tampilan lelang pengguna 2

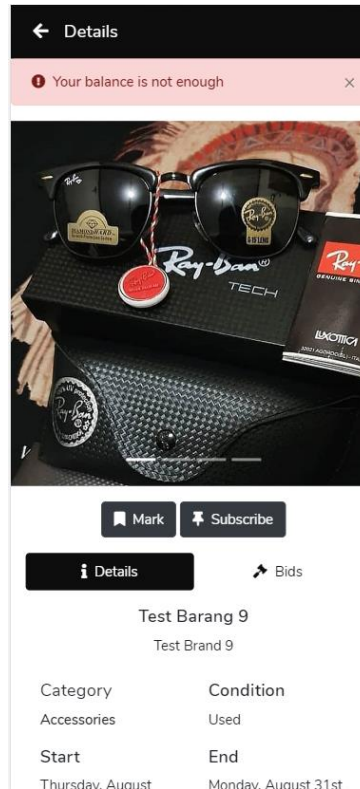
Bid & run

Untuk mengurangi praktik *bid & run*, kami mengimplementasikan sistem *punishment* pada sistem lelang *online* Auctentik. Sistem *punishment* yang dimaksud adalah berupa penahanan nominal yang sudah didepositkan oleh pengguna, sebelum pengguna dapat mengikuti lelang, pengguna harus melakukan deposit uang sejumlah nominal *open bid* barang yang dilelang untuk dapat mengikuti lelang barang tersebut seperti ditunjukkan pada Gambar 4.26. Apabila nominal saldo akun pengguna kurang dari nominal *open bid* barang yang dilelang, maka pengguna tidak dapat mengikuti lelang barang tersebut seperti yang ditunjukkan pada Gambar 4.27. Sebaliknya, apabila nominal saldo akun pengguna sama dengan atau lebih dari nominal *open bid* barang yang dilelang, maka pengguna dapat terdaftar pada lelang tersebut dan dapat mengikuti lelang barang yang bersangkutan seperti pada Gambar 4.28. Nominal deposit tersebut akan dikembalikan ke saldo pengguna apabila pengguna tidak memenangkan lelang, apabila pengguna memenangkan lelangnya namun tidak melakukan

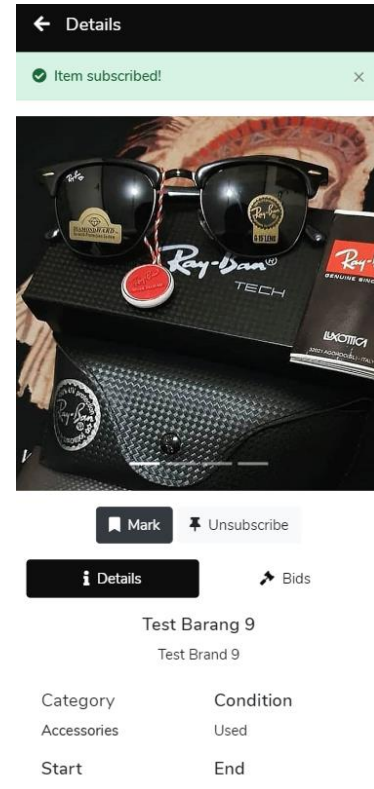
pelunasan pembayaran dalam kurun waktu 2 x 24 jam atau bisa disebut melakukan praktik *bid & run*, maka nominal deposit tersebut akan ditahan, sehingga diharapkan dapat menimbulkan efek jera dan mengurangi praktik *bid & run*.



Gambar 4.26 Peringatan ketika mendaftar lelang



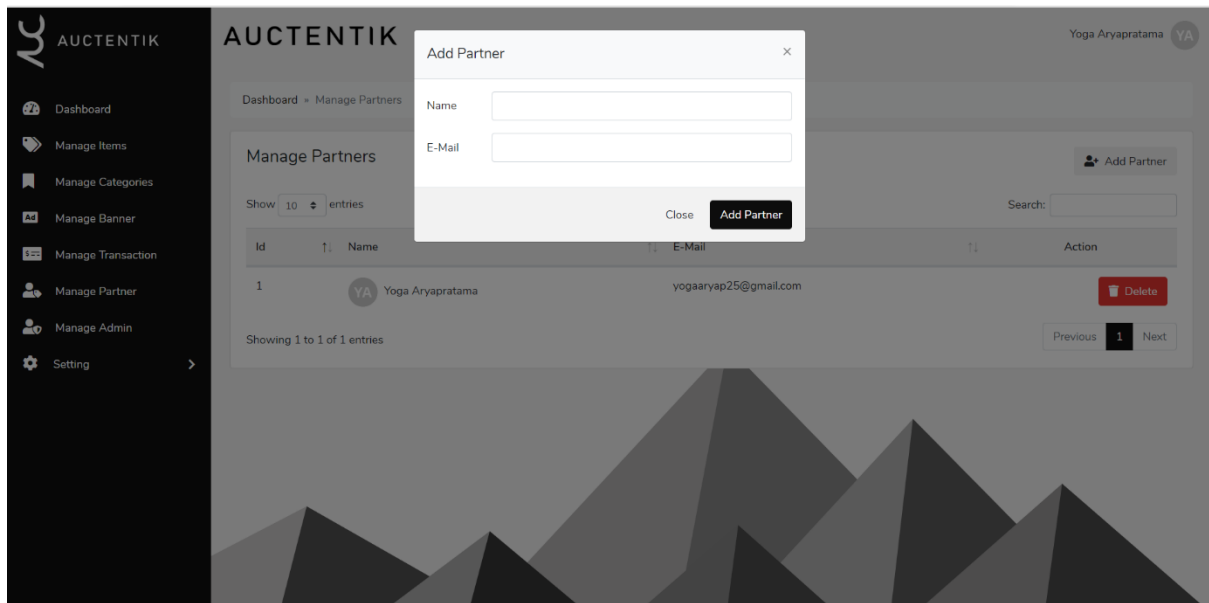
Gambar 4.27 Gagal mendaftar lelang



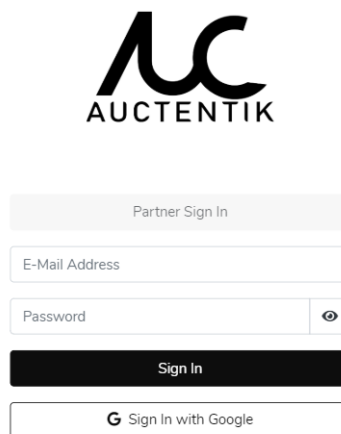
Gambar 4.28 Berhasil mendaftar lelang

Doubt of authenticity

Doubt of authenticity adalah masalah yang sangat kerap ditemui apabila melakukan transaksi secara *online*. Kami menawarkan solusi untuk paling tidak dapat mengurangi masalah tersebut, yaitu dengan membatasi pelelang yang dapat melelang barang pada sistem Auctentik. Untuk dapat melelang barang pada sistem Auctentik pelelang harus diverifikasi terlebih dahulu oleh tim Auctentik termasuk barang yang akan dilelang, setelah itu pelelang akan dibuatkan akun partner oleh admin agar dapat melelang barangnya pada sistem Auctentik seperti ditunjukkan pada Gambar 4.29, sehingga pada halaman autentikasi partner hanya ada opsi untuk masuk ke sistem, dan tidak ada opsi untuk mendaftar seperti halnya pada halaman autentikasi pengguna seperti ditunjukkan pada Gambar 4.30. Diharapkan dengan verifikasi pelelang dapat mengurangi keraguan akan orisinalitas pengguna.



Gambar 4.29 Halaman admin untuk menambah partner



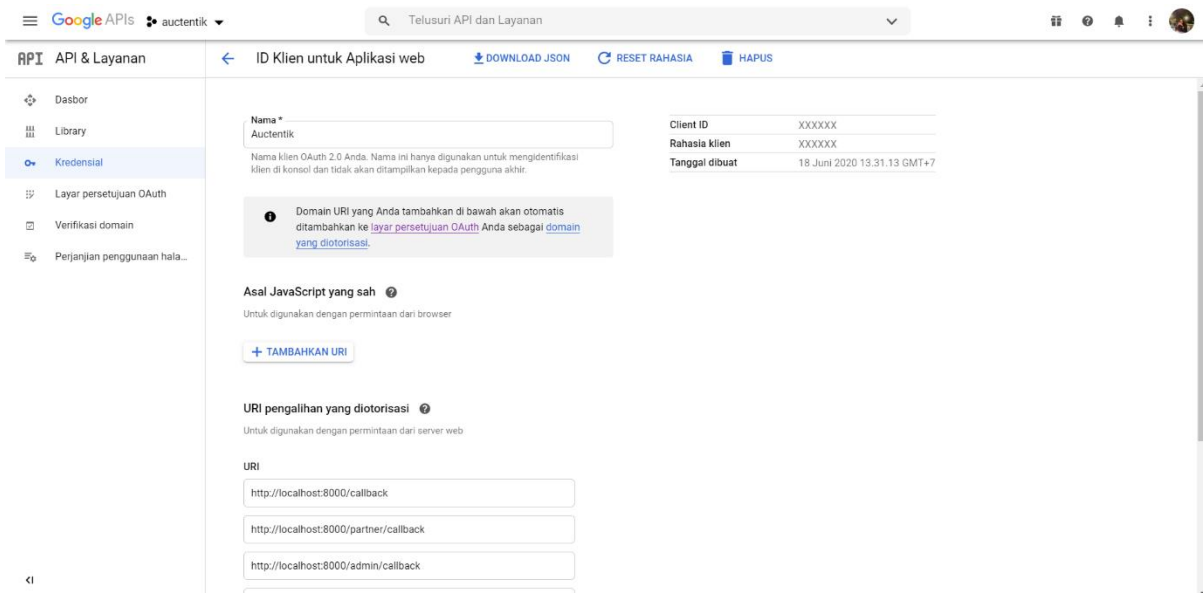
Gambar 4.30 Halaman *login* partner

4.2.2 Pengembangan Adaptif

Contoh kasus pengembangan adaptif pada pengembangan sistem Auctentik adalah ketika kami mengembangkan bagian autentikasi pengguna dan autentikasi partner. Rencana awal pengembangan bagian autentikasi baik untuk pengguna maupun untuk partner adalah

otentikasi secara manual dimana pengguna atau partner memasukkan *email* dan *password*, lalu dicocokkan dengan *email* dan *password* yang ada pada *database* sistem Auctentik. Kami mendapat masukan bahwa perihal autentikasi sekarang sudah banyak yang menggunakan *SSO* atau *single sign on*, dikarenakan dapat memudahkan dari sisi pengguna sehingga tidak perlu mengingat banyak akun pada banyak sistem, setelah dipertimbangkan kami memutuskan untuk menambahkan Google *SSO* pada sistem Auctentik. Penggunaan *framework* Laravel juga mempermudah kami dalam melakukan pengembangan adaptif, yang dalam kasus ini yaitu menambahkan Google *SSO* di autentikasi pengguna dan partner pada sistem Auctentik.

Pertama, kami memasang *package* Laravel Socialite, dengan menjalankan *command* `composer require laravel/socialite` pada *command line*, setelah itu kami membuat *project authentication* baru pada halaman *developer* Google seperti ditunjukkan pada Gambar 4.31, lalu menambahkan baris kode *key* pada *file .env* seperti ditunjukkan pada Gambar 4.32.



Gambar 4.31 Tampilan halaman *developer* Google

```
...
GOOGLE_CLIENT_ID=XXXXXX
GOOGLE_CLIENT_SECRET=XXXXXX
GOOGLE_REDIRECT_URL=http://localhost:8000/callback
...
```

Gambar 4.32 Baris kode pada *file .env*

Selanjutnya kami menambahkan fungsi untuk *men-handle login* di dalam *file* `AuthController.php` pada *folder* `app/Http/Controllers/User` seperti ditunjukkan pada Gambar 4.33.

```

...
public function redirectToProvider()
{
    return Socialite::driver('google')->redirectUrl(route('sso.google.callback'))->redirect();
}

public function handleProviderCallback()
{
    try {
        $user = Socialite::driver('google')->redirectUrl(route('sso.google.callback'))->user();
    } catch (\Exception $e) {
        return redirect()->route('showLoginForm');
    }

    if (User::where('google_id', $user->id)->exists()) {
        Auth::loginUsingId((User::where('google_id', $user->id)->first()->id);
    } elseif (User::where('email', $user->email)->exists()) {
        $currentUser = User::where('email', $user->email)->first();
        $currentUser->avatar = $user->getAvatar();
        $currentUser->google_id = $user->getId();
        $currentUser->save();
        Auth::loginUsingId($currentUser->id);
    } else {
        $newUser = User::create([
            'name' => $user->getName(),
            'email' => $user->getEmail(),
            'avatar' => $user->getAvatar(),
            'google_id' => $user->getId(),
        ]);
        Auth::loginUsingId($newUser->id);
    }

    return redirect()->intended(route('home'));
}
...

```

Gambar 4.33 Baris kode pada *file* AuthController.php

Terakhir kami menambahkan *routing* pada file routes/web.php seperti ditunjukkan pada Gambar 4.34.

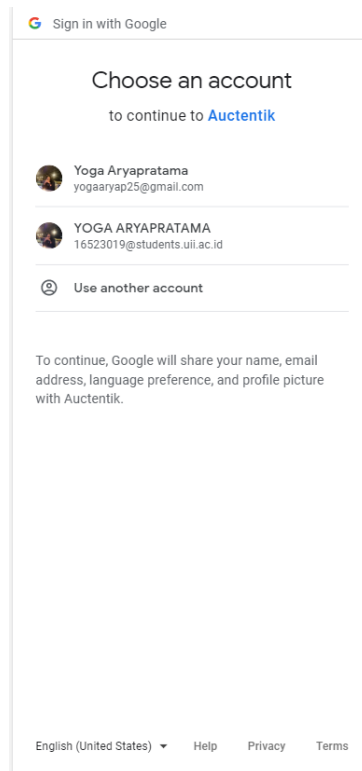
```

...
Route::get('/redirect', 'User/AuthController@redirectToProvider')->name('sso.google');
Route::get('/callback', 'User/AuthController@handleProviderCallback')->name('sso.google.callback');
...

```

Gambar 4.34 Baris kode pada *file* routes/web.php

Implementasi pada sistem Auctentik, pengguna dan partner dapat masuk ke sistem menggunakan akun Google tanpa harus membuat akun tersendiri pada sistem Auctentik, seperti ditunjukkan pada Gambar 4.35.



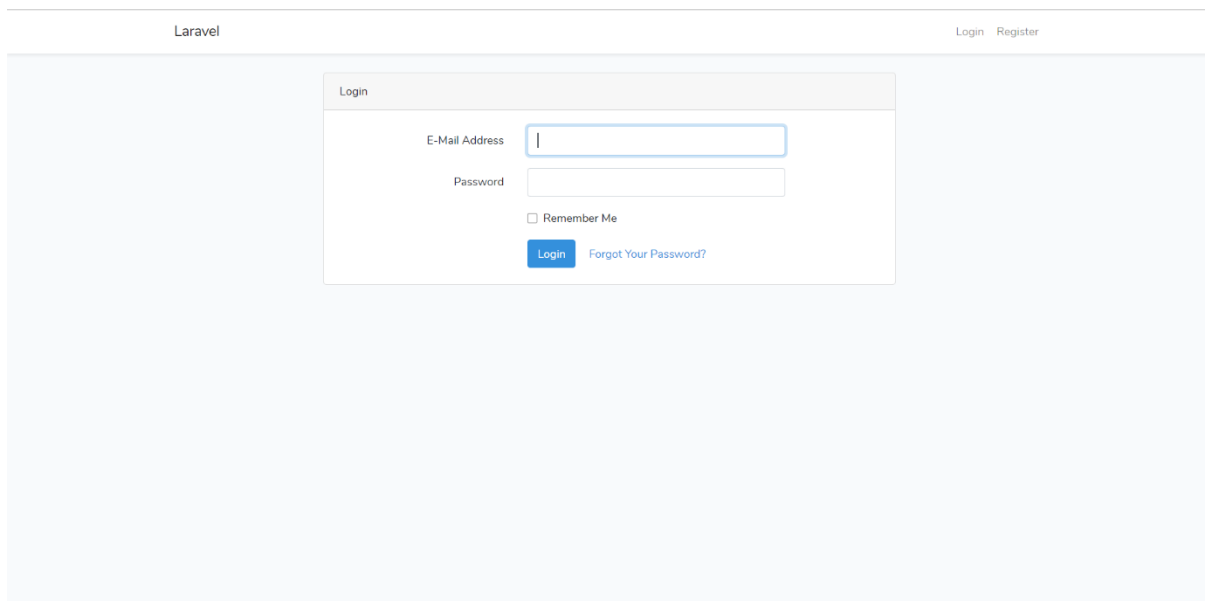
Gambar 4.35 Halaman tampilan otentikasi Google

4.2.3 Pengembangan Cepat

Digunakannya metode *Feature-driven Development* dan *framework* Laravel pada pengembangan ini diharapkan dapat mempercepat durasi pengembangan sistem Auctentik, dan setelah pengembangan tahap pertama sistem Auctentik selesai dikembangkan, dapat dilihat bahwa penggunaan metode *Feature-driven Development* dan *framework* Laravel mempercepat durasi pengembangan seperti ditunjukkan pada Tabel 4.1 apabila dibandingkan dengan *Gantt Chart* rencana pengembangan sistem Auctentik yang dapat dilihat pada Tabel 3.1. Pengembangan dapat dieksekusi dengan lebih cepat juga karena bantuan penggunaan *framework* Laravel, contohnya pada pengembangan autentikasi pengguna dan partner, menggunakan *framework* Laravel kami hanya menjalankan *command* `composer require laravel/ui "^1.0" --dev` dan `php artisan ui vue --auth` pada *command line*, dan dengan otomatis Laravel akan membuat *logic* dan tampilan autentikasi seperti ditunjukkan pada Gambar 4.36 dan Gambar 4.37.

Tabel 4.1 *Gantt Chart* realisasi pengembangan sistem Auctentik

#	Fitur	Durasi	Maret 2020			April 2020				
			18-20	23,24,26,27	30-3	6-9	13-17	20-24	27-30	
1	User	13 hari								
1.1	Authenticating a user	1 hari								
1.2	Bidding an item	4 hari								
1.3	Getting an item	2 hari								
1.4	Managing a balance	2 hari								
1.5	Managing a profile	4 hari								
2	Partner	14 hari								
2.1	Authenticating a partner	1 hari								
2.2	Managing an item	5 hari								
2.3	Shipping an item	2 hari								
2.4	Managing a balance	2 hari								
2.5	Managing a profile	4 hari								

Gambar 4.36 Tampilan halaman *login* bawaan Laravel

Laravel Login Register

Register

Name	<input style="width: 100%;" type="text"/>
E-Mail Address	<input style="width: 100%;" type="text"/>
Password	<input style="width: 100%;" type="password"/>
Confirm Password	<input style="width: 100%;" type="password"/>

[Register](#)

Gambar 4.37 Tampilan halaman *register* bawaan Laravel

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengembangan sistem Auctentik yang berbasis web menggunakan *framework* Laravel dengan metode *Feature-driven Development* (FDD) yang sudah dilakukan, maka dapat disimpulkan bahwa penggunaan metode *Feature-driven Development* dan *framework* Laravel dapat membuat pengembangan menjadi lebih cepat dalam masa pengembangannya dan dapat dengan mudah beradaptasi atau dilakukan ubahan sesuai dengan kondisi yang dibutuhkan seperti yang sudah dijabarkan pada Pembahasan.

5.2 Saran

Berdasarkan hasil pengembangan, maka perlu adanya pengembangan lebih lanjut pada sistem Auctentik untuk dapat menawarkan lebih dibandingkan dengan sistem lelang *online* yang ada saat ini. Seperti, pengembangan *native mobile apps*, optimasi *security* dan *performance*, dan penggunaan *payment gateway* untuk kemudahan pembayaran. Diperlukan juga adanya *maintenance* rutin terhadap sistem, agar sistem dapat berjalan sebagaimana mestinya.

DAFTAR PUSTAKA

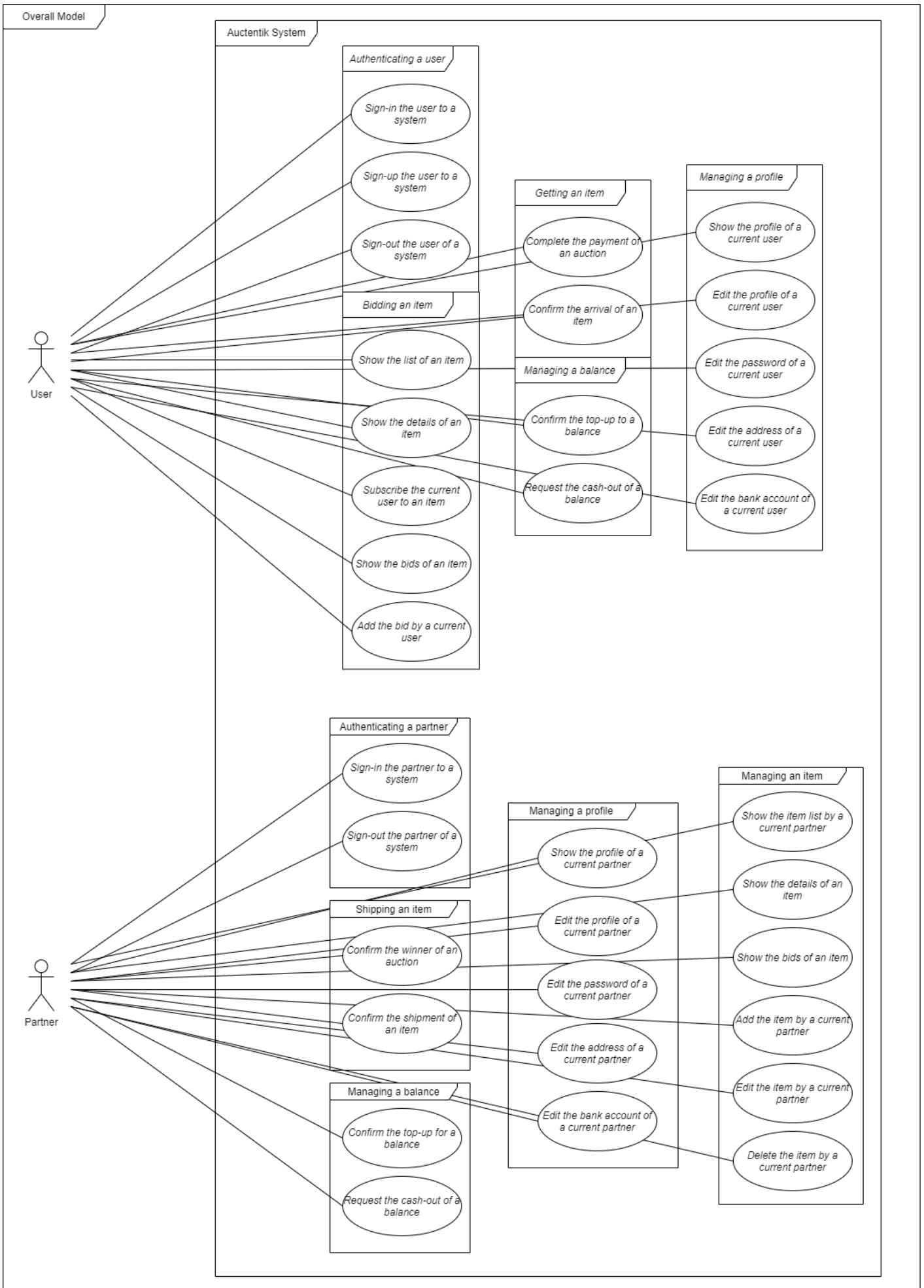
- Anthony, P. (2009). Overview of Electronic Auctions. In D. Mehdi Khosrow-Pour, *Encyclopedia of Information Science and Technology, Second Edition* (pp. 2953-2957).
- Arimbawa, I. W. (2014). Feature Driven Development (FDD), apakah bisa disebut Agile? .
- Ayertey, S., & Howell, K. (2019). Service Failure and Recovery Strategy in Computer-Mediated Marketing Environments (CMMEs). In G. Bowen, & W. Ozuem, *Leveraging Computer-Mediated Marketing Environments* (pp. 173-192).
- Azzura, S. N. (2019, March 11). *Per Februari 2019, Jumlah Startup RI Capai 2.070 Perusahaan*. Retrieved from merdeka.com: <https://www.merdeka.com/uang/per-februari-2019-jumlah-startup-ri-capai-2070-perusahaan.html>
- Bubendorfer, K., Palmer, B., & Welch, I. (2009). Trust and Privacy in Grid Resource Auctions. In E. Udoh, & F. Z. Wang, *Handbook of Research on Grid Technologies and Utility Computing: Concepts for Managing Large-Scale Applications* (pp. 85-96).
- Coad, P., Lefebvre, E., Luca, J. D., & Luca, E. D. (1999). *Java Modeling in Color with UML: Enterprise Components and Process*. USA: Prentice Hall PTR.
- Codignola, F., & Rancati, E. (2016). The Blending of Luxury Fashion Brands and Contemporary Art: A Global Strategy for Value Creation. In A. Vecchi, & C. Buckley, *Handbook of Research on Global Fashion Management and Merchandising* (pp. 50-76).
- David-West, O. (2016). E-Commerce Management in Emerging Markets. In I. Lee, *Encyclopedia of E-Commerce Development, Implementation, and Management* (pp. 200-222).
- Dr. Halim Barkatullah, S. M. (2017). *Hukum Transaksi Elektronik di Indonesia*. Bandung.
- Esquireme. (n.d.). Retrieved from Esquireme: <https://www.esquireme.com/content/26947-what-is-a-hypebeast-and-where-do-we-fnd-them>
- Feature Driven Development & Empirical Modelling. (2008). Retrieved from https://pdfs.semanticscholar.org/9453/601751dd145ec938c4820d185085a41294d6.pdf?_ga=2.172441702.450025743.1594523111-1277112924.1593408849
- Goyal, S. (2008). Major Seminar On Feature Driven Development. *Jennifer Schiller Chair of Applied Software Engineering*.
- Grant, M. (2020, March 10). *Startup Definition*. Retrieved from Investopedia: <https://www.investopedia.com/terms/s/startup.asp>

- Henry Ham, S. (2018, December 13). *Kelebihan Menggunakan Laravel Web Development*. Retrieved from School of Computer Science | BINUS University: <https://socs.binus.ac.id/2018/12/13/kelebihan-menggunakan-laravel-web-development/>
- Indrajit, R. E. (2012). *Evolusi E-Marketplace*.
- Jailani, N., Mukhtar, M., Al-aaidross, M. A., Patel, A., Abdullah, S., Yahya, Y., & Abdullah, Z. (2016). Agent-Based Auction E-Marketplace with Value Added Services and Islamic Shariah Compliance. In I. Lee, *Encyclopedia of E-commerce Development, Implementation, and Management (3 Volumes)* (pp. 888-904).
- Janita, M. S. (2016). Factors Contributing to Success in B2B E-Marketplaces. In I. Lee, *Encyclopedia of E-Commerce Development, Implementation, and Management* (pp. 944-953).
- Jayani, D. H. (2019, October 10). *Tren Pengguna E-Commerce Terus Tumbuh*. Retrieved from Databoks: <https://databoks.katadata.co.id/datapublish/2019/10/10/tren-pengguna-e-commerce-2017-2023>
- Liberati, D. (2009). Networked Experiments in Global E-Science. In J. Salmons, & L. Wilson, *Handbook of Research on Electronic Collaboration and Organizational Synergy* (pp. 615-625).
- Manvi, S. S., & Bhajantri, L. B. (2009). Agent Based Product Negotiation Models in Mobile Commerce. In I. Lee, *Handbook of Research on Telecommunications Planning and Management for Business* (pp. 820-835).
- Maulana, A. A., Susanto, A., & K., D. P. (2019). Rancang Bangun Web Scraping Pada Marketplace di Indonesia. *Journal of Information System*.
- Merchant Machine. (2019, March 25). *Saturated Sectors: Finding Gaps In The Ecommerce Market In 2020*. Retrieved from Merchant Machine: <https://merchantmachine.co.uk/saturated-sectors/>
- Orloff, M., & Gwyn, R. (2017, March 16). *Hypebeast Culture*. Retrieved from C Magazine: <https://cmagazine.org/culture/hypebeast-culture/>
- Palmer, S. R., & Felsing, J. M. (2002). *A Practical Guide to Feature-driven Development*. Prentice Hall PTR.
- Peterson, R. (2008). E-Commerce Models and Consumer Concerns. In L. A. Tomei, *Encyclopedia of Information Technology Curriculum Integration* (pp. 259-264).

- Praditya, D. (2019, January 30). *Prediksi Perkembangan Industri Startup Indonesia setelah 2020*. Retrieved from Tech in Asia: <https://id.techinasia.com/prediksi-startup-indonesia-2020>
- Risman. (2015). *Auction Reform: Lelang Indonesia Menuju Era Baru*. Retrieved from <https://www.kemenkeu.go.id/media/4475/auction-reform.pdf>
- Robehmed, N. (2013, December 16). *What Is A Startup?* Retrieved from Forbes: <https://www.forbes.com/sites/natalierobehmed/2013/12/16/what-is-a-startup/#20afe13a4044>
- Sánchez, Á. J., & Sánchez, M. R. (2020). Audiovisual Storytelling Approaches in the Marketing of Luxury Clothing Brands. In V.-M. Margalina, & J. M. Lavín, *Management and Inter/Intra Organizational Relationships in the Textile and Apparel Industry* (pp. 263-283).
- StartupRanking. (n.d.). *Countries - With the top startups worldwide*. Retrieved from Startup Ranking: <https://www.startupranking.com/countries>
- Tan, Y. L., & Macaulay, L. (2008). Small Business Collaboration Through Electronic Marketplaces. In N. Kock, *Encyclopedia of E-Collaboration* (pp. 569-576).
- Wang, J., Yao, J., & Xing, R. (2016). A Chronicle of an Unsuccessful E-Commerce Pioneer. In I. Lee, *Encyclopedia of E-Commerce Development, Implementation, and Management* (pp. 134-147).

LAMPIRAN A

Overall Use Case Model



LAMPIRAN B

Gantt Chart detail pengembangan sistem Auctentik

