

**INVESTIGASI *SCRUM PRODUCT BACKLOG* PADA  
PENGEMBANGAN PERANGKAT LUNAK: IMPLEMENTASI  
PEMBANGUNAN SISTEM INFORMASI PUSAT KARIR UIN  
JAKARTA**

**SKRIPSI**



Oleh :

Wahyu Nur Arizky

11180910000054

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH  
JAKARTA**

**2022 M / 1443 H**

**INVESTIGASI SCRUM PRODUCT BACKLOG PADA  
PENGEMBANGAN PERANGKAT LUNAK: IMPLEMENTASI  
PEMBANGUNAN SISTEM INFORMASI PUSAT KARIR UIN  
JAKARTA**

**SKRIPSI**

Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Sarjana Komputer (S.Kom.)



Oleh :

Wahyu Nur Arizky

11180910000054

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI SYARIF HIDAYATULLAH**

**JAKARTA**

**2022 M / 1443 H**

## LEMBAR PERSETUJUAN

### INVESTIGASI *SCRUM PRODUCT BACKLOG* PADA PENGEMBANGAN PERANGKAT LUNAK: IMPLEMENTASI PEMBANGUNAN SISTEM INFORMASI PUSAT KARIR UIN JAKARTA

Skripsi

Sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom)

Oleh

Wahyu Nur Arizky

11180910000054

Menyetujui

Dosen Pembimbing I



**Dewi Kharani, M.Sc.**

NIP. 198205222011012009

Dosen Pembimbing II



**Husni Teja Sukmana, S.T., M.Sc, Ph.D**

NIP. 197710302001121003

Mengetahui,  
Ketua Program Studi Teknik Informatika



**Dr. Imam Marzuki Shofi, M.T.**

NIP. 197202052008011010

## LEMBAR PENGESAHAN

Skripsi berjudul “INVESTIGASI *SCRUM PRODUCT BACKLOG* PADA PENGEMBANGAN PERANGKAT LUNAK: IMPLEMENTASI PEMBANGUNAN SISTEM INFORMASI PUSAT KARIR UIN JAKARTA” yang ditulis oleh Wahyu Nur Arizky NIM 11180910000054 telah diujikan dalam sidang munaqasyah Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta pada Mei 2022. Skripsi ini telah diterima sebagai salah satu syarat memperoleh gelar Sarjana Komputer (S.Kom) pada Program Studi Teknik Informatika.

Jakarta, Mei 2022

### Tim Penguji,

Penguji 1

**Siti Umami Masrurroh, M.Sc.**  
NIP. 198208232011012013

Penguji 2

**Fenty Eka Muzayyana Agustin, M.Kom.**  
NIP. 197608052009122003

### Tim Pembimbing,

Dosen Pembimbing I

**Dewi Khairani, M.Sc.**  
NIP. 198205222011012009

Dosen Pembimbing II

**Husni Teja Sukmana, S.T., M.Sc, Ph.D**  
NIP. 197710302001121003

### Mengetahui,

Dekan Fakultas Sains dan Teknologi,

Ketua Program Studi Teknik Informatika,



**Nashrul Haidem, S.Si., M.T., Ph.D**  
NIP. 197106082005011005

**Dr. Imam Marzuki Shofi, M.T.**  
NIP. 19720205 200801 1 010

## PERNYATAAN ORISINALITAS

Dengan ini saya menyatakan bahwa:

1. Skripsi ini merupakan benar hasil karya asli saya yang diajukan untuk memenuhi salah satu persyaratan memperoleh gelar Strata 1 di UIN Syarif Hidayatullah Jakarta.
2. Semua sumber yang saya gunakan dalam penulisan ini telah saya cantumkan sesuai dengan ketentuan yang berlaku di UIN Syarif Hidayatullah Jakarta.
3. Apabila dikemudian hari terbukti karya ini bukan hasil saya sendiri atau merupakan hasil jiplakan karya orang lain, maka saya bersedia menerima sanksi yang berlaku di UIN Syarif Hidayatullah Jakarta.

Jakarta, 12 Mei 2022



Wahyu Nur Arizky

11180910000054

## PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI

Sebagai civitas akademika UIN Syarif Hidayatullah Jakarta. Saya yang bertanda tangan di bawah ini:

Nama : Wahyu Nur Arizky  
NIM : 11180910000054  
Program Studi : Teknik Informatika  
Fakultas : Sains dan Teknologi  
Jenis Karya : Skripsi

Demi pembuatan ilmu pengetahuan saya menyetujui untuk memberikan kepada UIN Syarif Hidayatullah Jakarta Hak Bebas Royalti Non Eksklusif (*Non Exclusive Royalti Free Right*) atas karya ilmiah yang berjudul:

### **INVESTIGASI SCRUM PRODUCT BACKLOG PADA PENGEMBANGAN PERANGKAT LUNAK: IMPLEMENTASI PEMBANGUNAN SISTEM INFORMASI PUSAT KARIR UIN JAKARTA**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini UIN Syarif Hidayatullah Jakarta berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data, merawat dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/ pencipta dan sebagai pemilik Hak Cipta. Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 12 Mei 2022



Wahyu Nur Arizky



Nama : Wahyu Nur Arizky  
Program Studi : Teknik Informatika  
Judul : INVESTIGASI *SCRUM PRODUCT BACKLOG* PADA  
PENGEMBANGAN PERANGKAT LUNAK:  
IMPLEMENTASI PEMBANGUNAN SISTEM  
INFORMASI PUSAT KARIR UIN JAKARTA

### ABSTRAK

Pengembangan perangkat lunak skala besar sistem informasi Pusat Karir membutuhkan pengerjaan yang singkat serta kebutuhan untuk merespon setiap isu development dengan cepat. Dengan menggunakan metode pengembangan *scrum agile* dapat dicari tahu bagaimana perubahan *product backlog* yang terjadi selama pengembangan. Hasil dari penelitian ini berupa artefak-artefak *scrum* serta daftar faktor perubahan *product backlog* dalam masa 6 sprint. Faktor paling tinggi terjadinya perubahan *product backlog* adalah adanya Perbaikan cacat & bug (15 kali), kemudian disusul dengan penyesuaian requirements dan pengurangan scope (5x). Hal ini dapat terjadi dikarenakan pendefinisian user stories yang sudah ada mengacu pada aplikasi yang sudah berjalan sekarang, sehingga perubahan *requirements (backlog)* minim terjadi.

**Kata kunci** : *scrum, agile, product backlog, Puskarier*

**Jumlah Pustaka** : 25 Jurnal, 2 E-book, 1 Tesis, 2 Website

Name : Wahyu Nur Arizky  
Study Program : Informatics Engineering  
Title : INVESTIGASI *SCRUM PRODUCT BACKLOG* PADA  
PENGEMBANGAN PERANGKAT LUNAK:  
IMPLEMENTASI PEMBANGUNAN SISTEM  
INFORMASI PUSAT KARIR UIN JAKARTA

### ABSTRACT

The development of large-scale Career Center information system software requires a short timeframe and the need to respond to any development issues quickly. By using the agile scrum development method, we can find out how the product backlog changes that occurred during development. The results of this research are in the form of Scrum artifacts and a list of factors for changing the product backlog over a period of 6 sprints. The highest factor in the occurrence of changes in the product backlog is the existence of defect & bug fixes (15 times), then followed by adjustment of requirements and reduction of scope (5x). This can happen because the definition of existing user stories refers to applications that are already running now, so that changes in requirements (backlog) are minimal.

**Keywords : scrum, agile, product backlog, Puskarier**

**Bibliography : 25 Journal, 2 E-book, 1 Thesis, 2 Website**





## KATA PENGANTAR

Syukur alhamdulillah atas segala puja dan puji kehadirat Allah SWT. yang telah memberikan Rahmat dan Hidayah-Nya sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Shawalat berserta salam senantiasa tercurah limpah kepada Nabi Muhammad SAW yang telah membawa kita sebagai umatnya menuju zaman penuh ilmu pengetahuan.

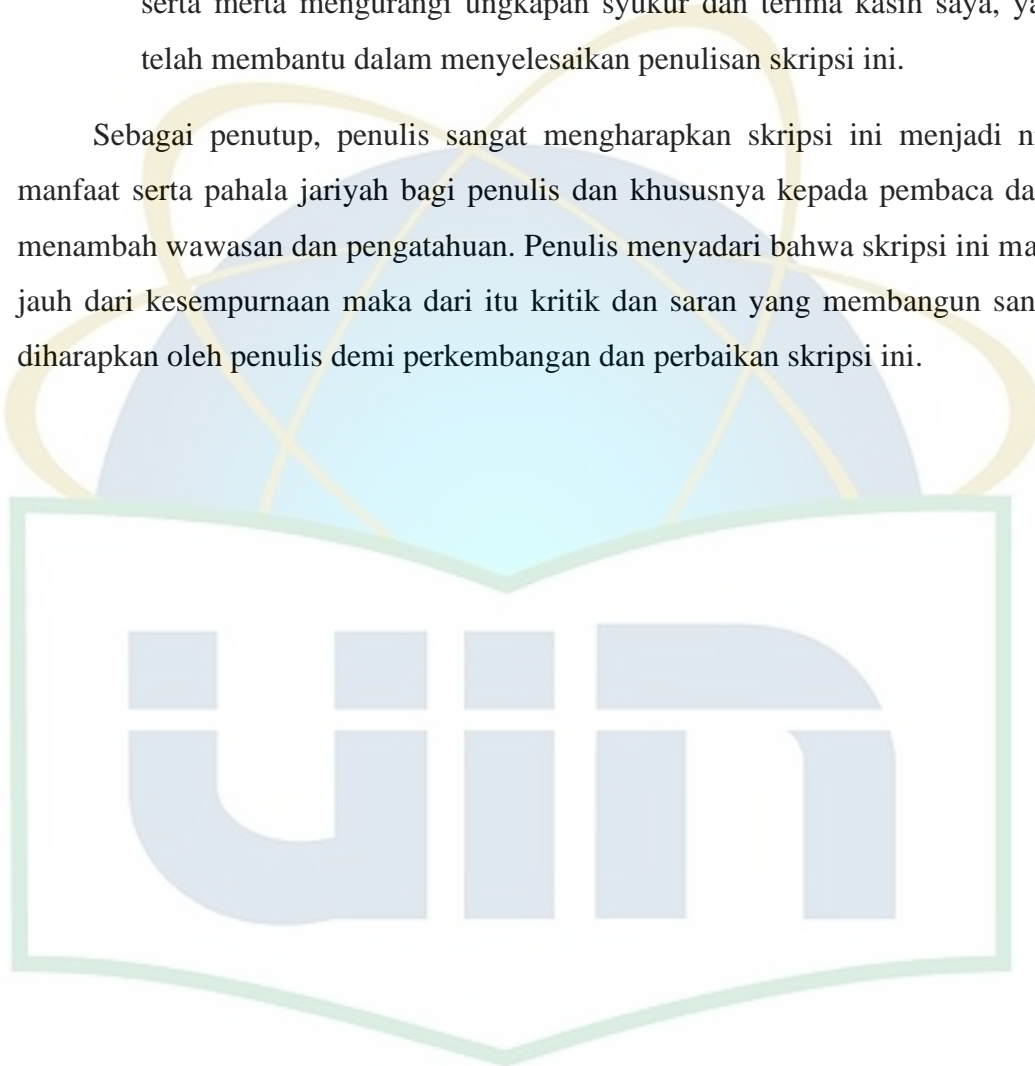
Skripsi yang berjudul “INVESTIGASI *SCRUM PRODUCT BACKLOG* PADA PENGEMBANGAN PERANGKAT LUNAK: IMPLEMENTASI PEMBANGUNAN SISTEM INFORMASI PUSAT KARIR UIN JAKARTA” yang telah disusun untuk memenuhi salah satu persyaratan untuk gelar sarjana computer Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Syarif Hidayatullah Jakarta.

Dalam proses menyelesaikan skripsi ini tentu tidak terlepas dari bantuan, dukungan, motivasi serta bimbingan, maka dari itu penulis ingin menyampaikan ucapan terima kasih kepada berbagai pihak:

1. Allah SWT. yang senantiasa mencurahkan nikmat sehat dan rahmat atas karunia kepada penulis.
2. Nashrul Hakiem, S.Si., M.T., Ph.D selaku Dekan Fakultas Sains dan Teknologi
3. Dr. Imam Marzuki Shofi, M.T selaku Ketua Program Studi Teknik Informatika.
4. Ibu Dewi Khairani, M.Sc. dan bapak Husni Teja Sukmana, S.T., M.Sc, Ph.D sebagai dosen pembimbing yang senantiasa sabar dan ikhlas dalam memberikan bimbingan, bantuan dan motivasi kepada penulis.
5. Ibu Salamah, selaku ketua pengelola Pusat Karier UIN Syarif Hidayatullah Jakarta yang telah memberikan kesempatan kepada penulis untuk melakukan penelitian pada Pusat Karier UIN Syarif Hidayatullah Jakarta.
6. Kedua orang tua penulis yang senantiasa mendukung, menasehati serta mendoakan penulis untuk sukses dan memudahkan dalam menyelesaikan setiap tahap dalam kehidupan. Serta kepada Kakak saya tercinta sebagai *support* tim yang membuat penulis termotivasi.

7. Seluruh teman-teman Teknik Informatika Angkatan 2018, yang secara langsung maupun tidak langsung sangat berpengaruh terhadap setiap kemajuan penulis baik dalam perkuliahan maupun kehidupan pergaulan sehari-hari.
8. Semua pihak yang tidak dapat penulis sebutkan satu persatu namun tidak serta merta mengurangi ungkapan syukur dan terima kasih saya, yang telah membantu dalam menyelesaikan penulisan skripsi ini.

Sebagai penutup, penulis sangat mengharapkan skripsi ini menjadi nilai manfaat serta pahala jariyah bagi penulis dan khususnya kepada pembaca dapat menambah wawasan dan pengetahuan. Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan maka dari itu kritik dan saran yang membangun sangat diharapkan oleh penulis demi perkembangan dan perbaikan skripsi ini.



## DAFTAR ISI

LEMBAR PERSETUJUAN .....	i
LEMBAR PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI .....	iv
ABSTRAK.....	v
ABSTRACT .....	vi
KATA PENGANTAR .....	vii
DAFTAR ISI .....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN.....	1
1. 1    Latar Belakang.....	1
1. 2    Identifikasi Masalah.....	5
1. 3    Rumusan Masalah.....	5
1. 4    Batasan Masalah .....	5
1. 5    Tujuan Penelitian .....	6
1. 6    Manfaat Penulisan.....	6
1. 7    Metode Penelitian .....	7
1. 8    Sistematika Penulisan .....	8
BAB II LANDASAN TEORI.....	9
2. 1    Investigasi .....	9
2. 2    Perangkat Lunak .....	9
2. 3    Pengembangan Perangkat Lunak .....	9
2.2.1    Model Waterfall.....	10
2.2.2    Model <i>Incremental Development</i> .....	11
2.2.3    V-Model.....	11
2.2.4    Spiral Development .....	12
2.2.5    Agile Development .....	13
2. 4    Scrum .....	15
2.3.1    Scrum Team.....	16
2.3.2    Scrum Events .....	17

2.3.3	Scrum Artefacts .....	18
2.5	Product Backlog.....	19
2.6	Pusat Karir UIN Jakarta.....	19
<b>BAB III METODOLOGI PENELITIAN .....</b>		<b>21</b>
3.1	Metode Pengumpulan Data.....	21
3.1.1	Observasi .....	21
3.1.2	Studi Pustaka.....	21
3.2	Metode Pengembangan Scrum .....	28
3.2.1	Scrum Team.....	28
3.2.2	Scrum Events .....	29
3.2.3	Scrum Artefacts .....	30
3.3	Metode Pengujian .....	30
3.4	Alur Penelitian .....	31
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>		<b>32</b>
4.1	Hasil Product Backlog .....	32
4.2	Review Sprint .....	42
4.2.1	Sprint Pertama.....	42
4.2.2	Sprint Kedua .....	45
4.2.3	Sprint Ketiga .....	48
4.2.4	Sprint Keempat .....	52
4.2.5	Sprint Kelima .....	55
4.2.6	Sprint Keenam .....	59
4.3	Perubahan Product Backlog.....	62
4.4	Analisa Perubahan Product Backlog.....	64
4.5	Hasil Pengujian UAT.....	66
<b>BAB V KESIMPULAN DAN SARAN .....</b>		<b>69</b>
5.1	Kesimpulan .....	69
5.2	Saran .....	69
<b>DAFTAR PUSTAKA .....</b>		<b>70</b>
<b>LAMPIRAN .....</b>		<b>73</b>

## DAFTAR GAMBAR

Gambar 1. Siklus Pengembangan Agile .....	2
Gambar 2. Enam tahap SDLC .....	10
Gambar 3. Model pengembangan V .....	12
Gambar 4. Model pengembangan spiral .....	13
Gambar 5. Model pengembangan <i>agile</i> .....	14
Gambar 6. Scrum events pada pengembangan puskarier .....	30
Gambar 7. Alur penelitian (Apoorva Srivastava, Sukriti Bhardwaj, 2017) .....	31
Gambar 8. Burndown chart sprint pertama .....	44
Gambar 9. Burndown chart sprint kedua .....	47
Gambar 10. Burndown chart sprint ketiga .....	50
Gambar 11. Burndown chart sprint keempat .....	53
Gambar 12. Burndown chart sprint kelima .....	57
Gambar 13. Burndown chart sprint keenam .....	60
Gambar 14. Perubahan backlog setiap sprint .....	64
Gambar 15. Perubahan product backlog selama pengembangan .....	65
Gambar 16. <i>Daily Standup</i> .....	73
Gambar 17. <i>Jira</i> .....	73
Gambar 18. <i>Figma</i> .....	74
Gambar 19. Tampilan Pencaker & Perusahaan .....	74
Gambar 20. Tampilan Admin .....	75
Gambar 21. Dokumentasi <i>API</i> menggunakan Postman .....	75
Gambar 22. <i>Gitlab</i> .....	76

## DAFTAR TABEL

Tabel 1. <i>Literature review</i> .....	22
Tabel 2. Scrum events pengembangan puskarier.....	29
Tabel 3. Hasil product backlog awal.....	32
Tabel 4. Product Backlog Awal setelah diurutkan berdasarkan priority .....	33
Tabel 5. Backlog child .....	35
Tabel 6. Sprint Planning Pertama .....	42
Tabel 7. Penambahan Backlog Sprint Pertama.....	43
Tabel 8. Review sprint pertama .....	44
Tabel 9. Sprint Planning Kedua .....	45
Tabel 10. Penambahan Backlog Sprint Kedua.....	46
Tabel 11. Review Sprint Kedua .....	47
Tabel 12. Sprint Planning Ketiga.....	48
Tabel 13. Penambahan Backlog Sprint Ketiga .....	49
Tabel 14. Review sprint ketiga .....	50
Tabel 15. Sprint Planning Keempat .....	52
Tabel 16. Penambahan Backlog Sprint Keempat.....	53
Tabel 17. Review sprint keempat.....	53
Tabel 18. Sprint Planning Kelima.....	55
Tabel 19. Penambahan Backlog Sprint Kelima .....	56
Tabel 20. Review sprint kelima .....	57
Tabel 21. Sprint Planning Keenam .....	59
Tabel 22. Penambahan Backlog Sprint Keenam.....	60
Tabel 23. Review sprint keenam.....	61
Tabel 24. Perubahan product backlog dalam pengembangan puskarier.....	63



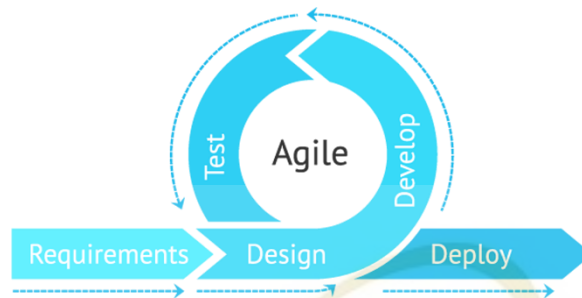
# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Pembangunan perangkat lunak merupakan proses untuk membuat perangkat lunak pada computer yang menggunakan satu atau lebih bahasa pemrograman tertentu dengan menyediakan fungsionalitas untuk menangani tujuan bisnis atau pribadi tertentu. Pengembangan perangkat lunak biasanya merupakan inisiatif yang direncanakan dengan sejumlah tahapan yang menghasilkan pembuatan perangkat lunak operasional (Bolung & Tampangela, 2017). Definisi pembangunan perangkat lunak secara umum adalah: “suatu proses dengan serangkaian aktivitas yang menciptakan produk perangkat lunak komputer, termasuk desain, pengembangan, pengujian, dan penerapannya.” Perangkat lunak itu sendiri adalah seperangkat instruksi atau program yang memberi tahu komputer apa yang harus dilakukan, dan instruksi ini tidak tergantung pada perangkat keras dan membuat komputer dapat diprogram.

Diantara banyak metode dalam pengembangan perangkat lunak, saat ini sudah banyak yang mengadopsi *agile* sebagai metode untuk pengembangan perangkat lunak. *Agile development* sendiri berarti kemampuan untuk menanggapi beberapa kondisi yang tidak pasti, oleh karena itu beberapa tindakan harus diambil untuk memastikan keberhasilan proyek pengembangan perangkat lunak



Gambar 1. Siklus Pengembangan Agile

Salah satu metode Agile yang mulai banyak digunakan saat ini adalah Scrum. Scrum adalah metodologi manajemen proyek untuk pengembangan perangkat lunak Agile yang menggunakan iterasi dan penambahan. Ini telah dirancang untuk mengelola persyaratan proyek yang berubah dengan cepat dengan meningkatkan komunikasi antara pengembang proyek, pemilik proyek, dan anggota tim lainnya. Pada tahun 1986, Hirotaka Takeuchi dan Ikujiro Nonaka menobatkan Scrum sebagai standar pengembangan produk baru di perusahaan otomotif dan produk konsumen (Lei et al., 2017).

Salah satu tahapan dari scrum adalah menentukan product backlog, product backlog pada dasarnya merupakan kumpulan (*list*) dari pekerjaan (cth: *user stories*, tugas-tugas, penyelesaian *bugs*) yang akan digunakan oleh tim untuk berkoordinasi dalam menyelesaikan sebuah proyek (Sedano et al., 2019). Pengembang (*developer*) biasanya menambahkan fitur baru, memodifikasi fitur yang sudah ada, dan memperbaiki *bugs* berdasarkan item teratas pada product backlog. karenanya, banyak tim menjadikan *backlog* sebagai acuan utama dalam mengarahkan pengembangan sebuah produk.

*Product backlog* akan diurutkan berdasarkan tingkatan-tingkatannya mulai dari yang paling penting hingga yang tidak terlalu penting. Tujuan diurutkannya product backlog ini pada dasarnya yakni untuk memudahkan tim untuk membagi list backlog. Berdasarkan literatur review yang telah dibuat oleh Ana Silva, mereka telah mengidentifikasi 8 faktor yang berbeda, dan mendapatkan bahwa *bussines value*, *dependency* (ketergantungan) dan *effort* (usaha) merupakan faktor yang

paling sering digunakan dalam mengurutkan tingkatan list *product backlog* (Silvax et al., 2017).

Memodifikasi backlog pasti terjadi dalam pengembangan. Menurut survei dari Ahmed Mubark Alsalemi, perubahan backlog merupakan hal yang umum dalam pengembangan dengan scrum, berdasarkan sureveinya alasan utama perubahan backlog karena *fixing defect* (memperbaiki cacat), menambahkan *requirements* yang kurang, dan peningkatan fungsionalitas (Alsalemi & Yeoh, 2016).

Implementasi scrum pada pengembangan produk juga telah dilakukan oleh Ndaru Ruseno 2019. Hasil dari penelitian ini adalah sebuah sistem reservasi online yang membantu mengelola sistem penyewaan fasilitas di Islamic Centre KH. Noer Alie dalam hal pemesanan, dan penyajian laporan. Selain itu aplikasi sistem reservasi ini dapat mengelola promo yang dapat berkembang dan berinovasi sepanjang waktu. Pembuatan aplikasi ini menggunakan bahasa pemograman PHP dan database MySQL (Ruseno, 2019).

Pengembangan produk dalam organisasi skala besar seperti universitas, tentunya akan banyak *stakeholder* yang terlibat dalam penentuan *product backlog*, ditambah organisasi besar pastinya akan membuat aplikasi yang lebih besar dan lebih banyak fitur dibanding organisasi kecil. Sehingga akan banyak tantangan yang di hadapi dalam mengimplementasikan scrum di organisasi besar.

Visi, misi dan tujuan UIN salah satunya adalah menghasilkan sarjana (lulusan) yang beriman, bertaqwa, dan berakhlak mulia serta memiliki keunggulan kompetitif dalam persaingan global; Salah satu program dalam mendukung visi tersebut adalah dengan membentuk sebuah Pusat Karir

Pusat Karir merupakan wadah sistem informasi pencarian kerja atau karir yang dimiliki UIN Jakarta. Misi pusat karir yakni Menjalankan pelayanan pengembangan karir yang unggul, inovatif, dan bermanfaat bagi mahasiswa dan alumni. Untuk mewujudkan dan menyukseskan misi tersebut maka dirasa perlu untuk membuat sistem perangkat lunak yang baik dan memudahkan sistem informasi ini berjalan.

Sebenarnya sudah ada website yang dimiliki oleh Pusat Karir yang menyajikan informasi untuk mahasiswa dan lulusan UIN Jakarta melalui lama <https://karir.uinjkt.ac.id> , namun website tersebut dirasa kurang cukup baik untuk dapat menjadi wadah yang bisa digunakan oleh mahasiswa, karena tidak ada interaksi secara langsung yang terjadi oleh pencari kerja dan penyedia kerja. Selain itu juga, aplikasi yang ada saat ini dikelola oleh developer luar yang tidak mudah untuk mendapatkan akses terhadap data secara langsung.

Berdasarkan analisis wawancara yang dilakukan oleh Annisa yang dilakukan dengan Ibu Salamah Agung M.A., Ph.D., selaku Kepala Pusat Karier UIN Syarif Hidayatullah Jakarta. Terdapat 3 masalah yang dialami Puskarier saat ini yakni, Pencari kerja belum dapat menerima notifikasi saat lamarannya diterima, Pencari kerja belum dapat melakukan assessment untuk mengetahui pekerjaan yang cocok tanpa harus melakukan konseling karier, Belum ada konseling karier untuk pencari kerja (Oktriyana, 2021).

Pengembangan perangkat lunak skala besar sistem informasi Pusat Karir membutuhkan pengerjaan yang singkat serta kebutuhan untuk merespon setiap isu development dengan cepat. Selain itu tim *development* yang mengembangkan adalah tim yang kecil, sehingga berdasarkan alasan diatas kami menggunakan metode scrum dalam pengembangannya.

Berdasarkan kondisi tersebut, mendorong peneliti untuk melakukan sebuah penelitian untuk menginvestigasi perubahan *product backlog* pada pengembangan sistem informasi Pusat Karir UIN Jakarta, sehingga dapat mengetahui bagaimana *product backlog* mengarahkan pekerjaan pengembangan dalam organisasi skala besar dan apa saja perubahan backlog yang terjadi. Peneliti memberikan judul penelitian **“Investigasi Scrum Product Backlog pada Pengembangan Perangkat Lunak: Implementasi Pembangunan Sistem Informasi Pusat Karir UIN Jakarta”**.

## 1.2 Identifikasi Masalah

Berdasarkan latar belakang diatas sehingga identifikasi masalah yang ada adalah sebagai berikut:

- Fungsionalitas system merupakan salah satu factor yang sangat penting, kondisi Puskarier saat ini softwrenya adalah produk lain yang kadang tidak memenuhi kebutuhan stakeholder puskarier
- Setiap perubahan yang dibutuhkan terhadap system, harus disubmit ke tim developer non UIN yang biasanya melalui tahapan/waktu yang lama
- Tidak dapat mengelola data dengan independent

## 1.3 Rumusan Masalah

- Bagaimana aktivitas pengembangan fungsionalitas (istilah dalam scrum adalah product backlog) pada pengembangan produk dalam organisasi skala besar seperti universitas dengan menggunakan scrum?
- Bagaimana hasil dari investigasi terhadap perubahan product backlog yang terjadi selama masa pengembangan?

## 1.4 Batasan Masalah

- Metode pengembangan menggunakan kerangka kerja *scrum* dari *agile mindset*
- Objek penelitian adalah pengembangan produk sistem informasi Pusat Karir UIN Jakarta
- Tools yang digunakan menggunakan aplikasi open source jira yang memiliki fitur *task assignment*, *scrum boards*, *roadmaps* dan pelaporan seperti *burndown chart*.



## 1.5 Tujuan Penelitian

Menginvestigasi aktivitas-aktivitas product backlog dan bagaimana pengaruh perubahan-perubahan product backlog dalam pengembangan produk organisasi skala besar seperti universitas menggunakan kerangka kerja scrum dari *agile*.

## 1.6 Manfaat Penulisan

### • Bagi Penulis

- Penulis dapat mengaplikasikan pengetahuan akademis yang didapat selama perkuliahan pada penelitian ini.
- Menambah wawasan penulis dalam bidang manajemen proyek terutama dengan kerangka kerja scrum.
- Sebagai salah satu syarat dalam penyelesaian gelar Strata Satu (S1) program studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.

### • Bagi Universitas

- Menambah kumpulan skripsi dari salah satu mahasiswa Teknik Informatika Fakultas Sains dan Teknologi dalam bidang manajemen proyek dengan kerangka kerja scrum.
- Menjadi sebuah tolak ukur bagi universitas dalam menentukan keberhasilan dan kemampuan penulis dalam mengimplementasikan ilmu yang sudah didapatkan selama menempuh pendidikan perkuliahan di universitas.

### • Bagi Masyarakat

- Berguna dalam segi keilmuan sehingga dapat menambah wawasan bagi masyarakat umum.



## 1.7 Metode Penelitian

- **Metode Pengumpulan Data**

Penulis menggunakan metode pengumpulan data yang dilakukan dengan studi pustaka dan studi literatur, yaitu dengan mempelajari buku-buku serta jurnal sebagai referensi terkait dengan informasi penelitian.

- **Metode Implementasi**

Penulis akan mengajak beberapa orang untuk membentuk suatu tim Scrum dan memberikan penawaran kepada Product Owner yang memiliki proyek yang ingin dikerjakan. Selama menjalani proyek ini akan dikerjakan keseluruhan tahap yang telah dipahami pada proses studi literatur, mencatat setiap Backlog yang dibutuhkan berupa: Nama Backlog, Detail, Estimasi, Prioritas, Tanggal dimulai, Tanggal selesai, Sumber Daya, dan Status. Kemudian pada setiap akhir iterasi akan dicatat persentase progress yang diperoleh. Proses ini secara incremental akan diulang sampai produk akhir selesai dibuat. Akan dicatat apakah ekspektasi fungsionalitas yang sudah disepakati sesuai dengan produk akhir.

## **1.8 Sistematika Penulisan**

Sistematika penulisan yang akan dilakukan dalam penelitian ini terdiri dari beberapa bagian:

### **BAB 1 PENDAHULUAN**

Bab ini membahas hal umum dalam penelitian, seperti latar belakang dari dari sebuah permasalahan yang diangkat, tujuan penelitian, manfaat penelitian, rumusan masalah, batasan masalah, metodologi penelitian, dan sistematika penulisan.

### **BAB II LANDASAR TEORI**

Bab ini menjelaskan beberapa materi dan teori yang dibutuhkan dalam melaksanakan penelitian ini.

### **BAB III METODE PENELITIAN**

Bab ini menjelaskan tentang metode yang digunakan untuk mendapatkan data dan metode untuk pengembangan sistem yang telah dibuat serta kerangka berpikir tugas akhir ini.

### **BAB IV IMPLEMENTASI**

Bab ini menjelaskan tentang proses implementasi dari metode yang digunakan untuk menyelesaikan permasalahan penelitian.

### **BAB V HASIL DAN PEMBAHASAN**

Bab ini membahas tentang hasil yang telah didapat dari proses implementasi dan eksperimen yang telah dilakukan pada bab sebelumnya.

### **BAB VI PENUTUP**

Bab ini menjelaskan tentang kesimpulan dari hasil yang telah didapat dan menjawab semua pokok permasalahan yang dirancang, serta saran-saran yang digunakan untuk penelitian lebih lanjut.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Investigasi**

Investigasi adalah penyelidikan dengan mencatat atau merekam fakta melakukan peninjauan, percobaan, dan sebagainya, dengan tujuan memperoleh jawaban atas pertanyaan (tentang peristiwa, sifat atau khasiat suatu zat, dsb) (*KBBI Daring*, n.d.).

#### **2.2 Perangkat Lunak**

Perangkat lunak (*software*) adalah program yang berisi kumpulan instruksi untuk melakukan proses pengolahan data (Utami & Asnawati, 2015). Perangkat lunak sebagai penghubung antara manusia sebagai pengguna dengan perangkat keras komputer, berfungsi menerjemahkan bahasa manusia kedalam bahasa mesin sehingga perangkat keras komputer memahami keinginan pengguna dan menjalankan instruksi yang diberikan dan selanjutnya memberikan hasil yang diinginkan oleh manusia tersebut (Utami & Asnawati, 2015).

#### **2.3 Pengembangan Perangkat Lunak**

Metode pengembangan perangkat lunak (*Software development model process*) adalah pendekatan untuk Software Development Life Cycle (SDLC) yang menggambarkan urutan langkah-langkah yang harus diikuti saat mengembangkan proyek perangkat lunak (Hijazi et al., 2012).

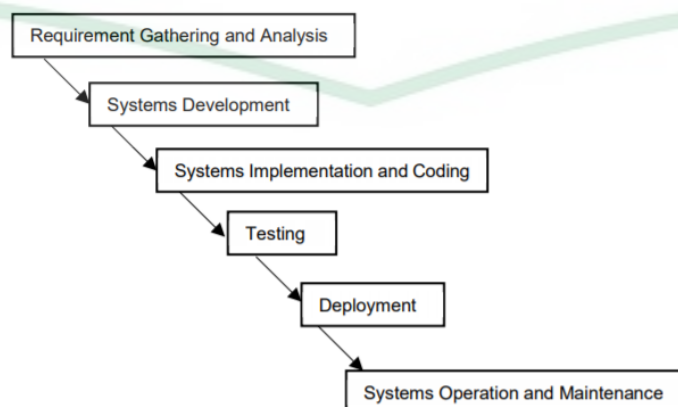
Banyak metodologi pengembangan perangkat lunak yang ada, mereka berbeda satu sama lain dalam hal waktu rilis, kualitas, dan manajemen risiko. Terlepas dari metodologi yang diikuti, aktivitas *basic lifecycle* terdapat dalam semua model *lifecycle*, tetapi mungkin dalam urutan yang berbeda. Model-model ini mungkin berurutan (misal *waterfall*) atau berulang (misal *evolutionary*). Mereka mungkin *specification-driven* (misal *waterfall*), *code-driven* (misal *evolutionary*),

atau *risk-driven* (misal spiral). Selain itu, mereka mungkin konvensional (misal *traditional waterfall*) atau *agile* (misal *scrum*) (Hijazi et al., 2012).

### 2.2.1 Model Waterfall

Pembangunan sistem secara keseluruhan dilakukan melalui beberapa tahapan/langkah. Metode pengembangan perangkat lunak dikenal juga dengan istilah *Software Development Life Cycle* (SDLC). Metode Waterfall merupakan metode pengembangan perangkat lunak tertua sebab sifatnya yang natural. Metode Waterfall merupakan pendekatan SDLC paling awal yang digunakan untuk pengembangan perangkat lunak. Urutan dalam Metode Waterfall bersifat serial yang dimulai dari proses perencanaan, analisa, desain, dan implementasi pada sistem (Bick et al., 2018).

Dengan mengikuti enam tahap SDLC, hasilnya adalah transisi yang mulus ke aplikasi yang akan dikembangkan. Semua input dari enam tahap meliputi berikut: 1) RUD (*Requirements Understanding Document*), Fungsional dan Persyaratan Bisnis. 2) HLD (*High-Level Design Document*) & LLD (*Low-Level Design Document*). 3) Aplikasi, kasus Uji Unit dan Hasil dari Pengembang. 4) Kasus Uji, Laporan Uji, Laporan Cacat, Log Masalah. 5) Panduan Pengguna, Definisi/Spesifikasi Lingkungan, Log Masalah. 6) Panduan Pengguna, Daftar *Production Tickets (Resolved)*, Daftar Fitur Baru yang Diimplementasikan.



Gambar 2. Enam tahap SDLC

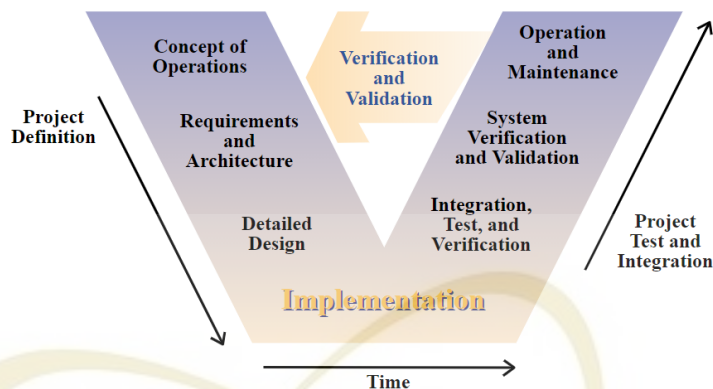
### **2.2.2 Model *Incremental Development***

Model pengembangan inkremental adalah varian dari model waterfall yang terdiri dari serangkaian siklus waterfall di mana proyek pengembangan perangkat lunak dipecah menjadi segmen yang lebih kecil yang disebut peningkatan (Hijazi et al., 2012). Usulan pengembangan inkremental adalah untuk mengakomodasi risiko yang melekat dari pelaksanaan proyek perangkat lunak secara keseluruhan selama siklus hidup tunggal dalam model waterfall murni (Department of the Air Force, 2003). Pertama-tama, karena proyek dipecah menjadi segmen-segmen yang lebih kecil, upaya pengembangan didistribusikan di antara beberapa peningkatan. Dengan demikian, risiko tersebar di beberapa iterasi daripada satu iterasi seperti dalam pengembangan air terjun murni. Jelas, akan lebih mudah untuk mengelola risiko tersebut dalam kasus sebelumnya.

### **2.2.3 V-Model**

V-Model adalah metodologi pengembangan linier unik yang digunakan selama siklus hidup pengembangan perangkat lunak (SDLC). V-Model berfokus pada metode mirip air terjun yang mengikuti tahapan langkah demi langkah yang ketat. Meskipun tahap awal adalah tahap desain yang luas, kemajuan berlanjut melalui tahap yang semakin terperinci, mengarah ke implementasi dan pengkodean, dan akhirnya kembali melalui semua tahap pengujian sebelum proyek selesai.





Gambar 3. Model pengembangan V

Model-V menyebarkan metode terstruktur dengan baik di mana setiap fase dapat diimplementasikan dengan dokumentasi terperinci dari fase sebelumnya. Aktivitas pengujian seperti perancangan pengujian dimulai pada awal proyek jauh sebelum pengkodean dan oleh karena itu menghemat banyak waktu proyek. Tujuan model V adalah untuk meningkatkan efisiensi dan efektivitas pengembangan perangkat lunak dan mencerminkan hubungan antara aktivitas pengujian dan aktivitas pengembangan seperti yang ditunjukkan pada Gambar 3. Model-V mungkin merupakan model paling tradisional yang diikuti untuk pengelolaan pengujian perangkat lunak (Mathur & Malik, 2010).

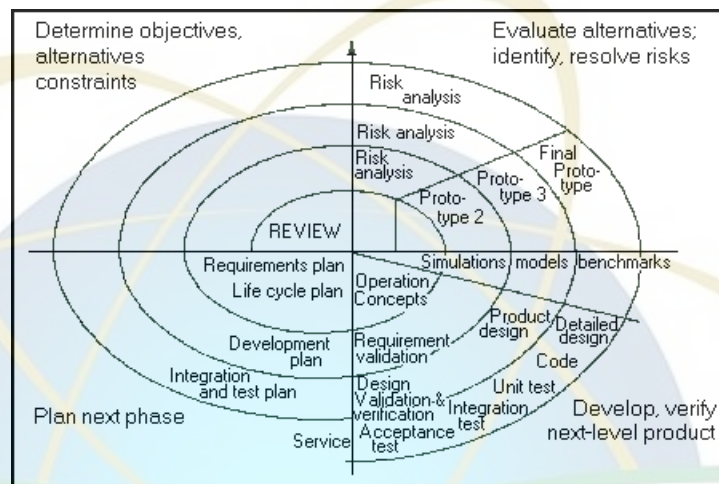
#### 2.2.4 Spiral Development

Model spiral diperkenalkan pertama kali oleh Barry Boehm pada makalahnya yang berjudul *Spiral Model of Software Development and Enhancement*. Barry Boehm menjelaskan bahwa model spiral merupakan model yang sangat berguna untuk melakukan pembangunan proyek-proyek besar dan prosesnya dilakukan dengan memperhatikan resiko proyek sehingga pada akhirnya akan menghasilkan model proses yang tepat sesuai kebutuhan pengguna.

Model Spiral adalah salah satu metode yang dapat digunakan dalam pengembangan perangkat lunak. Model spiral merupakan penggabungan



dari model prototyping dan model waterfall. Model prototyping yang fokus pada penyajian atau presentasi kepada user dengan format input dan output kemudian perangkat lunak akan dievaluasi. Model waterfall yang fokus kepada proses pengembangan perangkat lunak yang sistematis atau berurutan. Model spiral menekankan pada Analisa resiko setiap tahapannya (Boehm, 2007).



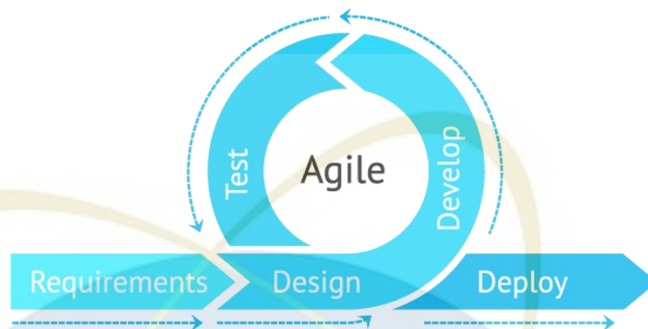
Gambar 4. Model pengembangan spiral

### 2.2.5 Agile Development

Agile adalah istilah yang pertama kali diperkenalkan pada tahun 2001 untuk merujuk pada sekelompok metodologi pengembangan perangkat lunak ringan yang berkembang pada pertengahan 1990-an termasuk Scrum (1995), Crystal Clear, Extreme Programming (1996), Pengembangan Perangkat Lunak Adaptif, Pengembangan Berbasis Fitur, dan Sistem Dinamis. Metode Pengembangan (DSDM) (1995) (Szalvay & Suite, 2004).

Berbeda dengan metodologi kelas berat (yaitu waterfall), metodologi Agile ringan tidak menekankan langkah proses formal; mereka melanjutkan proses pengembangan tanpa menunggu persyaratan formal dan spesifikasi desain. Poin utama yang menjadi fokus agile adalah komunikasi informal yang dekat antara pemangku kepentingan sistem yang berbeda termasuk pengembang dan perwakilan pelanggan. Memang, dalam agile,

komunikasi ini adalah sumber perencanaan, persyaratan, identifikasi risiko, umpan balik, dan perubahan (Hijazi et al., 2012).



Gambar 5. Model pengembangan *agile*

Agile memiliki pendekatan dan pandangan yang berbeda untuk setiap proyek dari metode tradisional air terjun dan prototipe dan dengan demikian membantu memenuhi kebutuhan proyek kontemporer dan selaras dengan tingkat perkembangan saat ini. Agile adalah metodologi di mana iterasi dan pengujian berkelanjutan berlangsung selama seluruh Siklus Hidup Pengembangan Perangkat Lunak (SDLC) suatu produk. Di antara metode Agile lainnya, seperti Metodologi Extreme Programming (EP), Metode Pengembangan Perangkat Lunak Dinamis (DSDM), Pengembangan Berbasis Fitur (FDD) dan lainnya, scrum adalah metode yang paling umum digunakan untuk pengembangan produk perangkat lunak (Meyer & Good, 2014).

Scrum meningkatkan keunggulannya sebagai metode Agile yang paling populer dari 40% yang dilaporkan dalam survei pertama pada tahun 2007 menjadi 56% sendiri dan 70% ketika dikombinasikan dengan metode lain pada tahun 2018. Pada saat yang sama, eXtreme Programming (XP) kehilangan pijakan dari menjadi metode terpopuler kedua (23%) yang digunakan dalam kombinasi dengan Scrum sebesar 6%. Sementara itu, Kanban sendiri (5%) dan dalam kombinasi dengan Scrum, Scrumban (8%), menggantikan DSDM (8%) (Hoda et al., 2018).

## 2.4 Scrum

Scrum adalah kerangka kerja yang digunakan terutama untuk pengembangan produk dan perangkat lunak. Sangat cocok untuk lingkungan yang kompleks di mana tim harus cepat bereaksi dan beradaptasi dengan situasi baru dalam sistem. Ini bukan metode yang cocok untuk lingkungan yang sederhana, jelas, jelas, jelas, mudah, stabil, dan dapat diprediksi. Kerangka kerja Scrum tidak hanya sangat efektif, tetapi juga merupakan cara yang produktif dan kreatif untuk memberikan produk akhir yang bernilai tinggi. Dengan hanya menggunakan beberapa aturan untuk pendekatan empiris, iteratif, dan inkrementalnya, Scrum berupaya mengendalikan risiko proses pengembangan produk dengan lebih baik. Scrum, meskipun cukup mudah untuk dipahami, terkadang bisa sangat menantang untuk dikuasai (Gonçalves, 2018).

Scrum, di sisi lain adalah jenis pendekatan Agile yang digunakan dalam pengembangan perangkat lunak. Selain Scrum, ada beberapa jenis pendekatan Agile lainnya, diantaranya Lean, Extreme Programming (XP), Crystal, Feature Driven Development dan Dynamic Systems Development Method (DSDM). Scrum sebagai metode Agile dalam praktik penting pada pengembangan perangkat lunak, dengan karakteristik sistem organisasi yang relatif fleksibel, umpan balik yang tepat waktu dan interaktif, manajemen datar yang berorientasi objektif dan partisipasi anggota yang efektif, dapat memecahkan kesulitan yang dihadapi oleh sebagian besar metode tradisional (Ma'arif et al., 2018). Liubchenko menjelaskan Scrum sebagai salah satu metode Agile yang lebih banyak digunakan. Ken Schwaber pertama kali mendeskripsikan Scrum pada tahun 1996 sebagai proses yang "menerima bahwa proses pengembangan tidak dapat diprediksi", memformalkan mentalitas "lakukan apa yang diperlukan", dan telah menemukan kesuksesan dengan banyak vendor perangkat lunak independen (Liubchenko, 2016).

Scrum telah menjadi wajah Agile Model dan istilah ini sering digunakan secara bergantian. Alasan popularitas scrum adalah karena memungkinkan pemilik produk untuk memulai proyek tanpa perencanaan sebelumnya yang ekstensif dan karena itu juga ada peningkatan signifikan dalam jumlah orang yang menggunakan Scrum sebagai teknik pengembangan dari laporan tahun 2013. Sekitar 82%

responden di antara 5000 peserta menggunakan Scrum dan 11% sedang menguji coba Scrum (Apoorva Srivastava, Sukriti Bhardwaj, 2017).

Terdapat 3 komponen utama dalam proses pengembangan menggunakan scrum, diantaranya Scrum Team, Scrum Events, dan Scrum Artefacts.

### 2.3.1 Scrum Team

Tiga anggota membentuk Tim Scrum dasar: Product Owner, Development Team (DT), dan Scrum Master. Mereka diharapkan dapat mengatur diri sendiri dan lintas fungsi tanpa bergantung pada orang lain di luar tim. Anggota Tim Scrum bekerja secara iteratif dan bertahap, memanfaatkan peluang untuk feedback yang ditawarkan oleh Scrum Events. Tim Scrum mewujudkan lima nilai keberanian, komitmen, keterbukaan, rasa hormat, dan fokus yang membantu membangun kepercayaan dengan semua yang terlibat. Anggota Tim Scrum belajar dan mengeksplorasi nilai-nilai tersebut saat mereka bekerja (Gonçalves, 2018).

**Scrum Master** adalah pemimpin Tim Scrum dan memastikan bahwa pekerjaan Tim Scrum telah dipahami dan ditindaklanjuti. Dia bertanggung jawab atas kepatuhan Tim Scrum terhadap aturan dan nilai-nilai Scrum serta prinsip-prinsip Agile. Scrum Master juga membantu orang-orang di luar Tim Scrum memahami kapan dan bagaimana mereka harus berinteraksi dengan Tim Scrum.

**Product Owner** bertanggung jawab untuk memaksimalkan kerja Tim Pengembang untuk menghasilkan produk bernilai tinggi. Product Owner juga mengelola Product Backlog, tugas yang melibatkan identifikasi secara jelas item Product Backlog, memprioritaskan item dalam Product Backlog untuk mencapai tujuan yang ditetapkan, mengoptimalkan nilai pekerjaan yang dilakukan oleh Tim Pengembang, dan memastikan bahwa Product Backlog transparan dan jelas bagi semua orang.

**Development Team** memberikan Increment produk yang berpotensi untuk dirilis di akhir setiap Sprint. Semua anggota Tim Pengembang dianggap sebagai pengembang, terlepas dari pekerjaan apa yang mereka lakukan. Hanya mereka yang dapat membuat Increment. Tim

Pengembang mengatur dirinya sendiri, karena tidak menerima instruksi atau saran dari siapa pun tentang cara mengubah Product Backlog menjadi Peningkatan fungsionalitas yang berpotensi dirilis.

### 2.3.2 Scrum Events

Scrum menggunakan time-box dalam pengembangan proyek dan fase perencanaan proyek. events di Scrum dirancang untuk memeriksa artefak dan mengadaptasi metode baru untuk memecahkan masalah proyek. Tujuan dari events ini adalah untuk memungkinkan transparansi, adaptasi, dan inspeksi dalam proses pengembangan yang menunjukkan komponen dari setiap Scrum Event (Lei et al., 2017).

untuk terciptanya komunikasi yang baik dalam pengembangan, terdapat 5 aktivitas (events) yang membantu pekerjaan secara efektif dan efisien. diantaranya Sprint Planning, Daily Scrum, Sprint Review, Sprint Restrospective, Sprint.

Sprint Planning atau perencanaan Sprint merupakan acara untuk membahas item mana yang akan dikerjakan selama Sprint berjalan. Dalam Scrum event ini, Product Owner akan bertugas untuk memastikan bahwa semua peserta acara siap untuk membahas item Product Backlog mana yang paling penting, serta memetakannya sesuai dengan Product Goal. Mereka mendiskusikan item product backlog apa yang harus dikerjakan selama sprint tertentu. Daftar item baru ini disebut dengan istilah sprint backlog.

Daily Scrum adalah acara yang dilakukan setiap hari untuk beberapa tujuan seperti :

- Memeriksa progress yang telah terjadi untuk mencapai Sprint Goals
- Menyesuaikan Sprint Backlog. (Sprint Backlog adalah istilah untuk kumpulan item dari Product Backlog yang dipilih untuk dikerjakan dalam Sprint)
- Menyesuaikan rencana kerja yang akan datang

Daily Scrum diadakan dengan durasi sekitar 15 menit dan ditujukan untuk tim Developer. Namun jika Product Owner dan Scrum Master juga



secara aktif mengerjakan item yang ada di Sprint Backlog, maka mereka dapat berpartisipasi sebagai Developer.

Sprint Review menjadi salah satu Sprint Events yang perlu dilakukan. Acara ini dilakukan di akhir Sprint untuk memeriksa bagaimana hasil dari Sprint yang telah dilakukan dan menentukan adaptasi yang diperlukan di waktu mendatang. Di dalam acara ini, Scrum Team akan mempresentasikan hasil kerja mereka kepada para stakeholders dan membahas bagaimana progress untuk mencapai Product Goal.

Sprint Retrospective menjadi sebuah pertemuan yang dapat digunakan oleh Scrum Team untuk memeriksa dirinya sendiri dan membuat perencanaan mengenai peningkatan kualitas dan efektivitas yang akan dilakukan pada Sprint berikutnya. Pada pertemuan ini, Scrum Team akan membahas apa saja yang sudah berjalan dengan baik selama Sprint, masalah apa yang dihadapi, dan bagaimana masalah tersebut diselesaikan atau bahkan tidak terselesaikan.

Scrum Events terakhir adalah Sprint. Sprint merupakan acara Scrum berdurasi tetap sekitar 1 bulan atau kurang untuk menciptakan suatu konsistensi. Sprint yang baru akan dimulai dengan segera setelah Sprint sebelumnya berakhir. Perlu Anda ketahui, Sprint ini dapat dikatakan sebagai “jantung” dalam kerangka kerja Scrum. Sprint juga menjadi “wadah” untuk acara-acara yang lain karena semua pekerjaan yang dibutuhkan untuk mencapai Product Goal, seperti Sprint Planning, Daily Scrum, Sprint Review, dan Sprint Retrospective terjadi di dalam Sprint ini.

### **2.3.3 Scrum Artefacts**

Scrum Artifact merupakan bagian dari kerangka kerja Scrum untuk merepresentasikan pekerjaan atau nilai bisnis guna terciptanya transparansi informasi. Artefak yang dijabarkan di dalam kerangka kerja Scrum ini dirancang sedemikian rupa untuk memaksimalkan transparansi informasi utama sehingga setiap orang memiliki pemahaman yang sama mengenai artefak tersebut (Gonçalves, 2018).



Scrum terdiri dari 3 artifact, yaitu :

- Product Backlog : untuk melihat progress untuk mencapai Product Goal
- Sprint Backlog : untuk melihat progress dari Sprint Goal
- Product Increment : untuk melihat progress yang dicapai menuju definisi “selesai”.

## 2.5 Product Backlog

Product backlog (atau hanya backlog) adalah list item pekerjaan (misalnya, user story, bug yang beredar, berbagai tugas) yang digunakan oleh tim perangkat lunak untuk mengoordinasikan pekerjaan yang harus dilakukan. Pengembang biasanya membangun fitur baru, memodifikasi fitur yang ada, dan memperbaiki bug berdasarkan item di bagian atas backlog. Bagi banyak tim, backlog adalah artefak manajemen proyek utama atau artefak non-kode utama. Siapa pun yang mengisi dan memprioritaskan backlog mengarahkan pengembangan produk (Sedano et al., 2019).

*Product backlog* akan diurutkan berdasarkan tingkatan-tingkatannya mulai dari yang paling penting hingga yang tidak terlalu penting. Tujuan diurutkannya product backlog ini pada dasarnya yakni untuk memudahkan tim untuk membagi list backlog. Berdasarkan literatur review yang telah dibuat oleh Ana Silva, mereka telah mengidentifikasi 8 faktor yang berbeda, dan mendapatkan bahwa *bussines value*, *dependency* (ketergantungan) dan *effort* (usaha) merupakan faktor yang paling sering digunakan dalam mengurutkan tingkatan list *product backlog* (Silvax et al., 2017).

## 2.6 Pusat Karir UIN Jakarta

Pusat Karier UIN Syarif Hidayatullah Jakarta resmi didirikan melalui Surat Keputusan Rektor pada November 2016. Pusat Karier secara struktur bertanggung jawab langsung kepada Rektor melalui koordinasi dengan Wakil Rektor Bidang

Kemahasiswaan dan Wakil Rektor Bidang Kerjasama. Layanan-layanan yang ingin diberikan, berupa konseling karier, job fair, campus hiring, tracer study, training soft skills, workshop, bursa kerja online, dan informasi magang (<https://karir.uinjkt.ac.id>, 2021).



## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Metode Pengumpulan Data**

Dalam pengumpulan data untuk penelitian ini, dilakukan penelitian terlebih dahulu guna mendapatkan data-data dan informasi yang terkait dengan sistem Puskarier untuk mengimplementasikan kerangka kerja *scrum*. Adapun teknik pengumpulan data yang dilakukan adalah observasi dan studi pustaka menggunakan literatur review.

##### **3.1.1 Observasi**

Observasi yang dilakukan untuk penelitian ini dilakukan dengan cara observasi langsung. Observasi langsung dilakukan dalam rangka mengamati sistem informasi Pusat Karier UIN Syarif Hidayatullah Jakarta yang sekarang. Observasi dilakukan dengan mengecek fungsi dari setiap fitur pada sistem yang ada, baik itu sebagai admin, perusahaan, maupun pencari kerja.

##### **3.1.2 Studi Pustaka**

Pada studi pustaka, penulis mencari data-data pendukung yang *relevan* dengan objek yang akan diteliti. Peneliti mendapatkan studi literatur dari berbagai sumber yaitu:

- 2 Buku
- 25 Jurnal Ilmiah
- 1 Tesis
- 2 Website

Penulis juga membuat literature review berdasarkan jurnal yang didapat.

Tabel 1. *Literature review*

No.	Judul	Tahun	Metode	Hasil
1.	Systematic literature review on agile practices in global software development (Vallon et al., 2018)	2018	<i>agile</i> secara umum	memberikan wawasan terperinci tentang keberhasilan penerapan praktik agile di GSD dari 1999 hingga 2016 dan juga mengidentifikasi praktik agile yang paling sering diterapkan dan skenario distribusi yang dilaporkan
2.	Adopting Scrum as an Agile Approach in Distributed Software Development: A Review of Literature (University of Mauritius et al., 2017)	2017	<i>scrum</i>	Berdasarkan studi penelitian yang telah dikumpulkan dalam penelitian ini, telah disimpulkan bahwa Scrum, pendekatan <i>Agile</i> dapat diadopsi dalam proyek DSD. Selanjutnya, penelitian menunjukkan bahwa Scrum meningkatkan proses DSD dengan mengatasi beberapa tantangan (koordinasi, komunikasi dan kerjasama) yang dihadapi di lingkungan DSD, melalui pelaksanaan pertemuan rutin ( <i>daily scrum</i> ) di antara anggota tim.
3.	The Scrum Product Backlog as a Tool for Steering the Product Development in a Large-Scale Organization (Seikola, 2010)	2010	<i>scrum</i>	Selain mengidentifikasi <i>stakeholder</i> dan kebutuhan data, tesis ini juga menunjukkan tiga temuan terkait bidang yang perlu ditingkatkan. Pertama, banyak <i>stakeholder</i> saat ini menuntut jaminan simpanan produk mereka sendiri. Kedua, beberapa serah terima

				dilakukan. Ketiga, proses inovasi terisolasi.
4.	SCRUM Model for Agile Methodology (Apoorva Srivastava, Sukriti Bhardwaj, 2017)	2017	<i>scrum</i>	<p>Scrum memiliki kemampuan untuk memenuhi kebutuhan ribuan proyek dan dapat ditingkatkan ke jumlah yang besar. Ini adalah aspek kecil dari implementasinya yang mengalami masalah signifikan seperti kurangnya otomatisasi dalam alur kerja dan pengujian, dukungan manajemen yang lebih baik, dan tidak tersedianya pekerja tim yang berpengalaman.</p> <p>Model scrum yang diusulkan bertujuan untuk membuat scrum pendekatan yang lebih komprehensif dibandingkan metodologi <i>agile</i> lainnya. Dengan perlakuan terpisah untuk tugas tradisional dan inovatif, siklus scrum keseluruhan terbukti lengkap dalam hal fungsionalitas dan dapat mengakomodasi perubahan dalam setiap iterasi. Peningkatan scrum sangat penting untuk pengguna saat ini dan semua langkah ini dapat lebih meningkatkan pengerjaan scrum. Ini tidak hanya dapat menjadi metodologi pengembangan perangkat lunak tetapi juga sebagai proses yang dapat</p>



				masuk ke dalam bidang apa pun yang disesuaikan untuknya dan tanpa perangkat dapat memberikan hasil yang efektif dan tahan lama.
5.	Ordering the Product Backlog in Agile Software Development Projects: A Systematic Literature Review (Silvax et al., 2017)	2017	<i>scrum</i>	<p>Peneliti telah mengidentifikasi 8 faktor yang berbeda. Diantaranya, nilai bisnis (11 studi), ketergantungan (8 studi) dan usaha (7 studi) adalah yang paling banyak dilaporkan, yang menunjukkan bahwa mereka adalah faktor berharga untuk dipertimbangkan dalam proses pembuatan product backlog. Selain itu, Peneliti menyimpulkan tidak ada konsensus tentang teknik mana yang akan diterapkan. Hasil penelitian ini dapat dijadikan acuan oleh para praktisi untuk memulai karya-karya baru, karena menyajikan gambaran kajian-kajian literatur tentang penataan backlog di ASD. Selanjutnya, berdasarkan hasil kami, kami menyimpulkan topik ini perlu penyelidikan lebih lanjut dengan melakukan studi empiris untuk menilai hasil penerapan praktik ini di lingkungan nyata.</p>
6.	The Product Backlog (Sedano et al., 2019)	2019	<i>scrum</i>	<i>Result:</i> Peneliti mengamati 13 praktik dan 6 kendala terkait pembuatan product backlog.

				<p>Keterbatasan: Grounded Theory tidak mendukung generalisasi statistik. Sementara teori backlog produk yang diusulkan tampaknya dapat diterapkan secara luas, organisasi dengan budaya pengembangan perangkat lunak yang berbeda dapat menggunakan praktik yang berbeda.</p> <p>Kesimpulan: Product backlog secara bersamaan merupakan model pekerjaan yang harus dilakukan dan objek batas yang membantu menjembatani kesenjangan antara proses menghasilkan user stories dan mewujudkannya dalam kode kerja. Ini muncul dari sensemaking (tim memahami konteks proyek) dan koevolusi (proses kognitif di mana tim secara bersamaan menyempurnakan pemahamannya tentang konteks bermasalah dan konsep solusi yang masih muda).</p>
7.	Product backlog rating: a case study on measuring test quality in scrum (Kayes et al., 2016)	2016	<i>scrum</i>	<p>Peneliti mengusulkan metrik untuk mengevaluasi proses pengujian di Scrum. Peneliti mencoba menawarkan metrik yang memberikan skor numerik untuk kualitas proses pengujian di Scrum. Peneliti telah membahas proses Scrum</p>

				<p>terlebih dahulu, menekankan pengujian di Scrum. Peneliti telah memanfaatkan pengalaman implementasi Scrum yang sukses dari Software People untuk menggambarkan proses Scrum. PBR metrik yang kami usulkan didasarkan pada kompleksitas Product Backlog dan penilaian penilaian pengujian. Metrik dapat memberikan kerangka kerja tambahan untuk manajemen mutu. Peneliti menyediakan studi kasus dari SoftwarePeople untuk menunjukkan bagaimana metrik digunakan dalam praktik.</p>
8.	Handling Transition Product Backlog with Scrum off Scrum (Reddaiah, 2016)	2016	<i>scrum</i>	Scrum off scrum menangani backlog semacam itu untuk melepaskan produk tanpa kesulitan.
9.	A Survey on Product Backlog Change Management and Requirement Traceability in Agile (Scrum) (Alsalemi & Yeoh, 2016)	2015	<i>scrum</i>	Perubahan Requirement lumrah terjadi di proyek Scrum karena 11 alasan. Alasan utamanya adalah memperbaiki cacat, persyaratan yang hilang, dan peningkatan fungsionalitas. Karena sifat metodologi pengembangan perangkat lunak Scrum yang fleksibel, Peneliti tidak dapat mengontrol frekuensi perubahan, tetapi dapat

				<p>merekam perubahan ini dan efeknya dengan lebih baik. Saat persyaratan berubah, sebagian besar praktisi menulis persyaratan yang diperbarui sebagai persyaratan baru (56,2%). Kecenderungan ini menunjukkan bahwa alih-alih mengelola kebutuhan yang ada dan mencatat alasan perubahan atau motivasi perubahan, praktisi hanya mencatatnya sebagai persyaratan baru dan mengembangkan sprint baru. Praktek ini memecahkan masalah langsung, tetapi tidak mendukung kegiatan manajemen proyek. Karena praktik ini, refactoring kode dan masalah kualitas perangkat lunak lainnya muncul. Versioning backlog produk tidak dikelola oleh sebagian besar praktisi (62,9%). Menurut praktisi, mereka mengubah backlog produk yang ada saat persyaratan berubah. Hanya 28,1% praktisi yang membuat versi baru dari product backlog karena terjadi perubahan pada persyaratan.</p>
10.	SCRUM Adoption: A Solution to Backlog Problems (Khan, 2016)	2016	<i>scrum</i>	<p>Satu kasus dipilih dan solusi yang diusulkan diimplementasikan, namun ada kebutuhan untuk</p>

				<p>menerapkan solusi yang diusulkan dalam banyak kasus dengan skenario yang berbeda, dengan tim yang lebih besar dan proyek yang kompleks. Di masa depan, teknik penentuan prioritas product backlog dapat diterapkan untuk memprioritaskan PO. Beberapa bagian dari solusi yang diusulkan yang dilewati selama studi kasus ini juga perlu diimplementasikan dan dievaluasi. Perusahaan target kami adalah warehouse kecil yang menerapkan "solusi yang kami usulkan" dengan versi uji coba alat SCRUM. Jika versi premium alat yang berbayar digunakan dengan fungsi prioritas otomatis, hasilnya bisa berbeda dan lebih fleksibel. Alat saat ini (versi <i>trial</i>) mendukung prioritas, namun bekerja secara manual di mana pengguna dapat melewati nilai bisnis.</p>
--	--	--	--	--

### 3.2 Metode Pengembangan Scrum

Pada pengembangan perangkat lunak menggunakan metode scrum ini, penulis mendefinisikan berdasarkan tiga bagian penting scrum, yakni Scrum Team, Scrum Events dan Scrum Artefacts.

#### 3.2.1 Scrum Team

Tim scrum pada penelitian ini diantaranya:



- Product Owner: Perwakilan Pusat Karier
- Scrum Master: Maarij
- Development Tim: Wahyu, Jabbar, Annisa, Fahmi

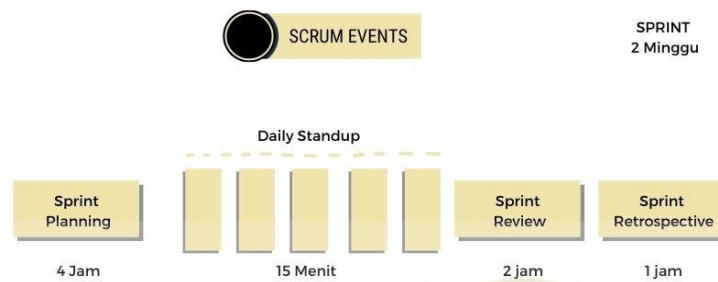
### 3.2.2 Scrum Events

Pada tahap awal pengembangan seluruh anggota tim berkumpul untuk merencanakan dan menentukan scrum events yang nantinya dijadikan acuan dan aturan untuk ditaati bersama. Bagian dari scrum events diantaranya, *sprint planning*, *daily standup*, *sprint*, *sprint review*, dan *sprint restrospective*.

Hasil dari penentuan bersama tersebut adalah sebagai berikut:

Tabel 2. Scrum events pengembangan puskarier

event	Deskripsi	Time Box
<b>Sprint Planning</b>	Diskusi menentukan sprint backlog yang akan dikerjakan	4 jam
<b>Daily Standup</b>	Diskusi setiap hari jam 19.00WIB untuk membahas secara singkat apa saja yang telah selesai dikerjakan, hambatan yang terjadi dan tugas yang akan dikerjakan selanjutnya	15 menit
<b>Sprint</b>	Tahap pengerjaan scrum dengan berdasarkan sprint planning yang telah dibuat	2 Minggu
<b>Sprint Review</b>	Diskusi setiap akhir sprint untuk membahas fitur mana yang sudah atau belum selesai dikerjakan semasa sprint 2 minggu	2 jam
<b>Sprint Retrospective</b>	Diskusi untuk membahas dan mengevaluasi performa tim selama pengembangan dalam satu sprint terakhir.	1 jam



Gambar 6. Scrum events pada pengembangan puskarier

Proses investigasi perubahan product backlog terjadi pada tahap sprint review dan sprint restrospective berupa perubahan apa saja yang terjadi selama satu masa sprint.

### 3.2.3 Scrum Artefacts

Pada pengembangan kali ini peneliti akan memaparan scrum artefacts dalam pengembangan menggunakan scrum aplikasi puskarier, diantaranya adalah, product backlog yang akan di list pada tahap awal pengembangan, kemudian perubahan-perubahan product backlog akan dicatat mulai dari awal hingga akhir sprint.

Kemudian, penulis akan mencatat sprint backlog apa saja yang dipilih dalam setiap sprint, sehingga dapat menjadi acuan dalam menganalisis data pada penelitian kali ini. Terakhir, penulis juga akan mencatat increment (peningkatan story points) yang terjadi setiap sprint.

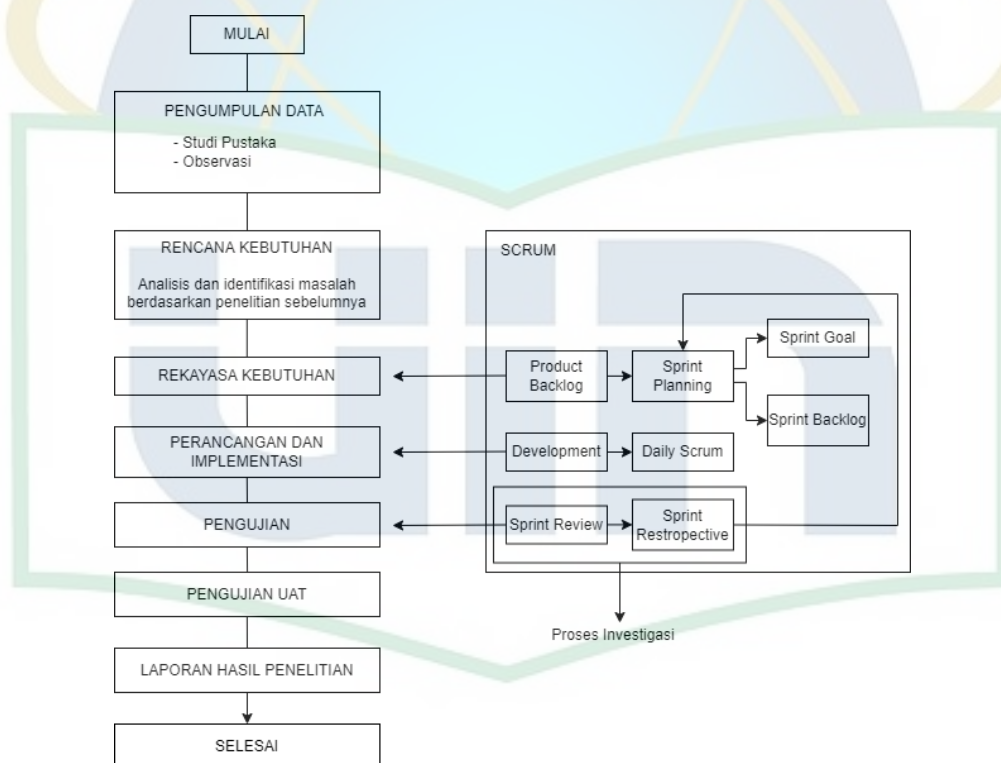
## 3.3 Metode Pengujian

Dalam pengembangan *software* ini, peneliti menggunakan metode UAT (User Acceptence Test) sebagai metode pengujian sistem. *User Acceptance Testing* (UAT) merupakan proses verifikasi bahwa solusi yang dibuat dalam sistem sudah sesuai untuk pengguna. Proses ini berbeda dengan pengujian sistem (memastikan software tidak crash dan sesuai dengan dokumen permintaan pengguna), melainkan memastikan bahwa solusi dalam sistem tersebut akan bekerja untuk pengguna

(yaitu, tes bahwa pengguna menerima solusi di dalam sistem). Tes penerimaan pengguna merupakan langkah penting sebagai baris terakhir verifikasi untuk memeriksa kesiapan pengiriman perangkat lunak yang sesuai dengan harapan pengguna (Leung & Wong, 1997).

Tipe UAT yang digunakan pada penelitian ini adalah *blackbox testing*. *Black-Box Testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. khususnya pada input dan output aplikasi (apakah sudah sesuai dengan apa yang diharapkan atau belum) (Mustaqbal et al., 2015).

### 3.4 Alur Penelitian



Gambar 7. Alur penelitian (Apoorva Srivastava, Sukriti Bhardwaj, 2017)

## BAB IV HASIL DAN PEMBAHASAN

### 4.1 Hasil Product Backlog

Penentuan Product Backlog ditentukan oleh diskusi dengan Product Owner (PO) yaitu perwakilan Pusat Karir UIN Jakarta. Isi dari product backlog atau selanjutnya kita sebut User Stories berisi id fitur, detail, story point dan tingkat priority. Satuan story point yang ditentukan adalah 1 story point senilai 2 jam, sedangkan tingkat priority yang ditentukan adalah low, medium dan high.

Tabel 3. Hasil product backlog awal

<i>ID</i> <i>Fitur</i>	<i>Sebagai</i>	<i>User Stories</i>	<i>Story</i> <i>Point</i>	<i>Priority</i>
1	Pencaker	Login dan registrasi	23	High
2	Pencaker	Melakukan pencarian	12	High
3	Pencaker	Melihat halaman detail pencari kerja	14	High
4	Pencaker	Melihat halaman perusahaan	14	High
5	Pencaker	Menandai lowongan kerja favorit	10	Medium
6	Pencaker	Melamar pekerjaan	10	High
7	Pencaker	Melihat lowongan kerja yang telah dilamar	14	High
8	Pencaker	Mengelola profile pencari kerja	24	High
9	Pencaker	Melakukan assessment	20	Low
10	Pencaker	Mengelola konseling kerja	20	Low
11	Pencaker	Melihat artikel	20	Medium
12	Pencaker	Melihat event	20	Medium
13	Pencaker	Daftar event	10	Medium
14	Pencaker	Mengelola event yang didaftar	12	Medium
15	Perusahaan	Mengelola profil perusahaan	20	High
16	Perusahaan	Mengelola lowongan kerja	18	High
17	Perusahaan	Mengelola data pelamar	24	High
18	Perusahaan	Melakukan kontrak berlangganan	14	Low
19	Perusahaan	Melihat riwayat kontrak berlangganan	10	Low
20	Alumni	Mengisi data tracer study	20	High
21	Admin	Mengelola akun pencaker	15	High
22	Admin	Mengelola akun perusahaan	15	High
23	Admin	Mengelola akun admin	15	High

24	Admin	Mengelola aktivitas akun	10	Low
25	Admin	Mengelola data industri pekerjaan	10	Medium
26	Admin	Mengelola data peran pekerjaan	10	Medium
27	Admin	Mengelola kategori artikel	10	Medium
28	Admin	Mengelola artikel	12	Medium
29	Admin	Mengelola lowongan kerja	16	High
30	Admin	Mengelola data pelamar	16	High
31	Admin	Mengelola data institusi pendidikan	10	Medium
32	Admin	Mengelola data bidang studi	10	Medium
33	Admin	Mengelola profil	20	High
34	Admin	Mengelola FAQ	10	Medium
35	Admin	Mengelola kategori event	10	Medium
36	Admin	Mengelola event	12	Medium
37	Admin	Mengelola pendaftar event	16	Medium
38	Admin	Mengelola data fakultas dan prodi	10	High
39	Admin	Mengelola kategori kuesioner tracer study	10	High
40	Admin	Mengelola kuesioner tracer study	26	High
41	Admin	Mengelola data tracer study	20	High
42	Admin	Mengelola database alumni	16	High
43	Admin	Mengelola data konselor karier	16	Low
44	Admin	Mengelola data pendaftar konseling kerja	16	Low
45	Admin	Mengelola data paket langganan	16	Low
46	Admin	Mengelola data pembayaran	16	Low
47	Admin	Melihat overview dan statistik	30	Medium
48	Admin	Mengelola deskripsi pusat karier	10	Medium

Kemudian, setelah product backlog selesai di definisikan, kita urutkan product backlog berdasarkan priority nya. Tujuannya adalah supaya memudahkan dalam menentukan sprint planning nantinya

Tabel 4. Product Backlog Awal setelah diurutkan berdasarkan priority

<i>ID</i>	<i>Sebagai</i>	<i>User Stories</i>	<i>Story Point</i>	<i>Priority</i>
<i>Fitur</i>				
1	Pencaker	Login dan registrasi	23	High
2	Pencaker	Melakukan pencarian	12	High
3	Pencaker	Melihat halaman detail pencari kerja	14	High



4	Pencaker	Melihat halaman perusahaan	14	High
6	Pencaker	Melamar pekerjaan	10	High
7	Pencaker	Melihat lowongan kerja yang telah dilamar	14	High
8	Pencaker	Mengelola profile pencari kerja	24	High
15	Perusahaan	Mengelola profil perusahaan	20	High
16	Perusahaan	Mengelola lowongan kerja	18	High
17	Perusahaan	Mengelola data pelamar	24	High
20	Alumni	Mengisi data tracer study	20	High
21	Admin	Mengelola akun pencaker	15	High
22	Admin	Mengelola akun perusahaan	15	High
23	Admin	Mengelola akun admin	15	High
29	Admin	Mengelola lowongan kerja	16	High
30	Admin	Mengelola data pelamar	16	High
33	Admin	Mengelola profil	20	High
38	Admin	Mengelola data fakultas dan prodi	10	High
39	Admin	Mengelola kategori kuesioner tracer study	10	High
40	Admin	Mengelola kuesioner tracer study	26	High
41	Admin	Mengelola data tracer study	20	High
42	Admin	Mengelola database alumni	16	High
5	Pencaker	Menandai lowongan kerja favorit	10	Medium
11	Pencaker	Melihat artikel	20	Medium
12	Pencaker	Melihat event	20	Medium
13	Pencaker	Daftar event	10	Medium
14	Pencaker	Mengelola event yang didaftar	12	Medium
25	Admin	Mengelola data industri pekerjaan	10	Medium
26	Admin	Mengelola data peran pekerjaan	10	Medium
27	Admin	Mengelola kategori artikel	10	Medium
28	Admin	Mengelola artikel	12	Medium
31	Admin	Mengelola data institusi pendidikan	10	Medium
32	Admin	Mengelola data bidang studi	10	Medium
34	Admin	Mengelola FAQ	10	Medium
35	Admin	Mengelola kategori event	10	Medium
36	Admin	Mengelola event	12	Medium
37	Admin	Mengelola pendaftar event	16	Medium
47	Admin	Melihat overview dan statistik	30	Medium
48	Admin	Mengelola deskripsi pusat karier	10	Medium
9	Pencaker	Melakukan assessment	20	Low

10	Pencaker	Mengelola konseling kerja	20	Low
18	Perusahaan	Melakukan kontrak berlangganan	14	Low
19	Perusahaan	Melihat riwayat kontrak berlangganan	10	Low
24	Admin	Mengelola aktivitas akun	10	Low
43	Admin	Mengelola data konselor karier	16	Low
44	Admin	Mengelola data pendaftar konseling kerja	16	Low
45	Admin	Mengelola data paket langganan	16	Low
46	Admin	Mengelola data pembayaran	16	Low

Kemudian Product backlog dipecah lagi oleh scrum master, agar memudahkan pembagian kepada Developer Team (DT).

Tabel 5. Backlog child

<b>FITUR</b>	<b>STORIES CHILD</b>	<b>STORY POINTS</b>	<b>PENANGGUNG JAWAB</b>
<b>FITUR 1</b>	Desain Fitur 1	6	Fahmi
	Membuat Backend API Fitur 1	5	Jabbar
	Implementasi UI Fitur 1	6	Wahyu
	Implementasi API ke UI Fitur 1	4	Wahyu
	Unit Testing Fitur 1	2	Annisa
<b>FITUR 2</b>	Desain Fitur 2	2	Fahmi
	Membuat API Fitur 2	2	Jabbar
	Implementasi UI Fitur 2	4	Wahyu
	Implementasi API ke UI Fitur 2	2	Wahyu
	Unit Testing Fitur 2	2	Annisa
<b>FITUR 3</b>	Desain Fitur 3	3	Fahmi
	Membuat API Fitur 3	3	Jabbar
	Implementasi UI Fitur 3	3	Wahyu
	Implementasi API ke UI Fitur 3	3	Wahyu
	Unit Testing Fitur 3	2	Annisa
<b>FITUR 4</b>	Desain Fitur 4	3	Fahmi
	Membuat API Fitur 4	3	Jabbar
	Implementasi UI Fitur 4	3	Wahyu
	Implementasi API ke UI Fitur 4	3	Wahyu
	Unit Testing Fitur 4	2	Annisa

<b>FITUR</b> <b>5</b>	Desain Fitur 5	2	Fahmi
	Membuat API Fitur 5	3	Jabbar
	Implementasi UI Fitur 5	2	Wahyu
	Implementasi API ke UI Fitur 5	2	Wahyu
	Unit Testing Fitur 5	1	Annisa
<b>FITUR</b> <b>6</b>	Desain Fitur 6	2	Fahmi
	Membuat API Fitur 6	2	Jabbar
	Implementasi UI Fitur 6	2	Wahyu
	Implementasi API ke UI Fitur 6	3	Wahyu
	Unit Testing Fitur 6	1	Annisa
<b>FITUR</b> <b>7</b>	Desain Fitur 7	3	Fahmi
	Membuat API Fitur 7	3	Jabbar
	Implementasi UI Fitur 7	3	Wahyu
	Implementasi API ke UI Fitur 7	3	Wahyu
	Unit Testing Fitur 7	2	Annisa
<b>FITUR</b> <b>8</b>	Desain Fitur 8	4	Fahmi
	Membuat API Fitur 8	6	Jabbar
	Implementasi UI Fitur 8	8	Wahyu
	Implementasi API ke UI Fitur 8	4	Wahyu
	Unit Testing Fitur 8	2	Annisa
<b>FITUR</b> <b>9</b>	Desain Fitur 9	2	Fahmi
	Membuat API Fitur 9	6	Jabbar
	Implementasi UI Fitur 9	6	Wahyu
	Implementasi API ke UI Fitur 9	4	Wahyu
	Unit Testing Fitur 9	2	Annisa
<b>FITUR</b> <b>10</b>	Desain Fitur 10	2	Fahmi
	Membuat API Fitur 10	6	Jabbar
	Implementasi UI Fitur 10	6	Wahyu
	Implementasi API ke UI Fitur 10	4	Wahyu
	Unit Testing Fitur 10	2	Annisa
<b>FITUR</b> <b>11</b>	Desain Fitur 11	3	Fahmi
	Membuat API Fitur 11	4	Jabbar
	Implementasi UI Fitur 11	7	Wahyu

	Implementsi API ke UI Fitur 11	4	Wahyu
	Unit Testing Fitur 11	2	Annisa
<b>FITUR 12</b>	Desain Fitur 12	3	Fahmi
	Membuat API Fitur 12	4	Jabbar
	Implementasi UI Fitur 12	5	Wahyu
	Implementsi API ke UI Fitur 12	6	Wahyu
	Unit Testing Fitur 12	2	Annisa
<b>FITUR 13</b>	Desain Fitur 13	2	Fahmi
	Membuat API Fitur 13	3	Jabbar
	Implementasi UI Fitur 13	2	Wahyu
	Implementsi API ke UI Fitur 13	2	Wahyu
	Unit Testing Fitur 13	1	Annisa
<b>FITUR 14</b>	Desain Fitur 14	2	Fahmi
	Membuat API Fitur 14	4	Jabbar
	Implementasi UI Fitur 14	3	Wahyu
	Implementsi API ke UI Fitur 14	2	Wahyu
	Unit Testing Fitur 14	1	Annisa
<b>FITUR 15</b>	Desain Fitur 15	4	Fahmi
	Membuat API Fitur 15	6	Jabbar
	Implementasi UI Fitur 15	5	Wahyu
	Implementsi API ke UI Fitur 15	3	Wahyu
	Unit Testing Fitur 15	2	Annisa
<b>FITUR 16</b>	Desain Fitur 16	4	Fahmi
	Membuat API Fitur 16	5	Jabbar
	Implementasi UI Fitur 16	4	Wahyu
	Implementsi API ke UI Fitur 16	3	Wahyu
	Unit Testing Fitur 16	2	Annisa
<b>FITUR 17</b>	Desain Fitur 17	4	Fahmi
	Membuat API Fitur 17	8	Jabbar
	Implementasi UI Fitur 17	6	Wahyu
	Implementsi API ke UI Fitur 17	4	Wahyu
	Unit Testing Fitur 17	2	Annisa
	Desain Fitur 18	0	Fahmi

<b>FITUR 18</b>	Membuat API Fitur 18	0	Jabbar
	Implementasi UI Fitur 18	0	Wahyu
	Implementsi API ke UI Fitur 18	0	Wahyu
	Unit Testing Fitur 18	0	Annisa
<b>FITUR 19</b>	Desain Fitur 19	0	Fahmi
	Membuat API Fitur 19	0	Jabbar
	Implementasi UI Fitur 19	0	Wahyu
	Implementsi API ke UI Fitur 19	0	Wahyu
<b>FITUR 20</b>	Unit Testing Fitur 19	0	Annisa
	Desain Fitur 20	5	Fahmi
	Membuat API Fitur 20	5	Jabbar
	Implementasi UI Fitur 20	4	Wahyu
<b>FITUR 21</b>	Implementsi API ke UI Fitur 20	4	Wahyu
	Unit Testing Fitur 20	2	Annisa
	Desain Fitur 21	3	Fahmi
	Membuat API Fitur 21	4	Jabbar
<b>FITUR 22</b>	Implementasi UI Fitur 21	3	Wahyu
	Implementsi API ke UI Fitur 21	3	Wahyu
	Unit Testing Fitur 21	2	Annisa
	Desain Fitur 22	3	Fahmi
<b>FITUR 23</b>	Membuat API Fitur 22	4	Jabbar
	Implementasi UI Fitur 22	3	Wahyu
	Implementsi API ke UI Fitur 22	3	Wahyu
	Unit Testing Fitur 22	2	Annisa
<b>FITUR 24</b>	Desain Fitur 23	3	Fahmi
	Membuat API Fitur 23	4	Jabbar
	Implementasi UI Fitur 23	3	Wahyu
	Implementsi API ke UI Fitur 23	3	Wahyu
<b>FITUR 24</b>	Unit Testing Fitur 23	2	Annisa
	Desain Fitur 24	0	Fahmi
	Membuat API Fitur 24	0	Jabbar
	Implementasi UI Fitur 24	0	Wahyu
	Implementsi API ke UI Fitur 24	0	Wahyu



	Unit Testing Fitur 24	0	Annisa
<b>FITUR 25</b>	Desain Fitur 25	2	Fahmi
	Membuat API Fitur 25	3	Jabbar
	Implementasi UI Fitur 25	2	Wahyu
	Implementasi API ke UI Fitur 25	2	Wahyu
	Unit Testing Fitur 25	1	Annisa
<b>FITUR 26</b>	Desain Fitur 26	2	Fahmi
	Membuat API Fitur 26	3	Jabbar
	Implementasi UI Fitur 26	2	Wahyu
	Implementasi API ke UI Fitur 26	2	Wahyu
	Unit Testing Fitur 26	1	Annisa
<b>FITUR 27</b>	Desain Fitur 27	2	Fahmi
	Membuat API Fitur 27	3	Jabbar
	Implementasi UI Fitur 27	2	Wahyu
	Implementasi API ke UI Fitur 27	2	Wahyu
	Unit Testing Fitur 27	1	Annisa
<b>FITUR 28</b>	Desain Fitur 28	2	Fahmi
	Membuat API Fitur 28	4	Jabbar
	Implementasi UI Fitur 28	2	Wahyu
	Implementasi API ke UI Fitur 28	2	Wahyu
	Unit Testing Fitur 28	2	Annisa
<b>FITUR 29</b>	Desain Fitur 29	4	Fahmi
	Membuat API Fitur 29	4	Jabbar
	Implementasi UI Fitur 29	3	Wahyu
	Implementasi API ke UI Fitur 29	3	Wahyu
	Unit Testing Fitur 29	2	Annisa
<b>FITUR 30</b>	Desain Fitur 30	4	Fahmi
	Membuat API Fitur 30	4	Jabbar
	Implementasi UI Fitur 30	3	Wahyu
	Implementasi API ke UI Fitur 30	3	Wahyu
	Unit Testing Fitur 30	2	Annisa
<b>FITUR 31</b>	Desain Fitur 31	2	Fahmi
	Membuat API Fitur 31	3	Jabbar

	Implementasi UI Fitur 31	2	Wahyu
	Implementasi API ke UI Fitur 31	2	Wahyu
	Unit Testing Fitur 31	1	Annisa
<b>FITUR 32</b>	Desain Fitur 32	2	Fahmi
	Membuat API Fitur 32	3	Jabbar
	Implementasi UI Fitur 32	2	Wahyu
	Implementasi API ke UI Fitur 32	2	Wahyu
	Unit Testing Fitur 32	1	Annisa
<b>FITUR 33</b>	Desain Fitur 33	4	Fahmi
	Membuat API Fitur 33	6	Jabbar
	Implementasi UI Fitur 33	5	Wahyu
	Implementasi API ke UI Fitur 33	3	Wahyu
	Unit Testing Fitur 33	2	Annisa
<b>FITUR 34</b>	Desain Fitur 34	2	Fahmi
	Membuat API Fitur 34	3	Jabbar
	Implementasi UI Fitur 34	2	Wahyu
	Implementasi API ke UI Fitur 34	2	Wahyu
	Unit Testing Fitur 34	1	Annisa
<b>FITUR 35</b>	Desain Fitur 35	2	Fahmi
	Membuat API Fitur 35	3	Jabbar
	Implementasi UI Fitur 35	2	Wahyu
	Implementasi API ke UI Fitur 35	2	Wahyu
	Unit Testing Fitur 35	1	Annisa
<b>FITUR 36</b>	Desain Fitur 36	2	Fahmi
	Membuat API Fitur 36	3	Jabbar
	Implementasi UI Fitur 36	4	Wahyu
	Implementasi API ke UI Fitur 36	2	Wahyu
	Unit Testing Fitur 36	1	Annisa
<b>FITUR 37</b>	Desain Fitur 37	2	Fahmi
	Membuat API Fitur 37	5	Jabbar
	Implementasi UI Fitur 37	4	Wahyu
	Implementasi API ke UI Fitur 37	3	Wahyu
	Unit Testing Fitur 37	2	Annisa

<b>FITUR 38</b>	Desain Fitur 38	2	Fahmi
	Membuat API Fitur 38	3	Jabbar
	Implementasi UI Fitur 38	2	Wahyu
	Implementasi API ke UI Fitur 38	2	Wahyu
	Unit Testing Fitur 38	1	Annisa
<b>FITUR 39</b>	Desain Fitur 39	2	Fahmi
	Membuat API Fitur 39	3	Jabbar
	Implementasi UI Fitur 39	2	Wahyu
	Implementasi API ke UI Fitur 39	2	Wahyu
	Unit Testing Fitur 39	1	Annisa
<b>FITUR 40</b>	Desain Fitur 40	4	Fahmi
	Membuat API Fitur 40	8	Jabbar
	Implementasi UI Fitur 40	6	Wahyu
	Implementasi API ke UI Fitur 40	6	Wahyu
	Unit Testing Fitur 40	2	Annisa
<b>FITUR 41</b>	Desain Fitur 41	3	Fahmi
	Membuat API Fitur 41	6	Jabbar
	Implementasi UI Fitur 41	5	Wahyu
	Implementasi API ke UI Fitur 41	4	Wahyu
	Unit Testing Fitur 41	2	Annisa
<b>FITUR 42</b>	Desain Fitur 42	2	Fahmi
	Membuat API Fitur 42	4	Jabbar
	Implementasi UI Fitur 42	4	Wahyu
	Implementasi API ke UI Fitur 42	4	Wahyu
	Unit Testing Fitur 42	2	Annisa
<b>FITUR 43</b>	Desain Fitur 43	2	Fahmi
	Membuat API Fitur 43	5	Jabbar
	Implementasi UI Fitur 43	4	Wahyu
	Implementasi API ke UI Fitur 43	4	Wahyu
	Unit Testing Fitur 43	1	Annisa
<b>FITUR 44</b>	Desain Fitur 44	2	Fahmi
	Membuat API Fitur 44	5	Jabbar
	Implementasi UI Fitur 44	4	Wahyu

<b>FITUR</b> <b>45</b>	Implementsi API ke UI Fitur 44	4	Wahyu
	Unit Testing Fitur 44	1	Annisa
	Desain Fitur 45	0	Fahmi
	Membuat API Fitur 45	0	Jabbar
	Implementasi UI Fitur 45	0	Wahyu
<b>FITUR</b> <b>46</b>	Implementsi API ke UI Fitur 45	0	Wahyu
	Unit Testing Fitur 45	0	Annisa
	Desain Fitur 46	0	Fahmi
	Membuat API Fitur 46	0	Jabbar
	Implementasi UI Fitur 46	0	Wahyu
<b>FITUR</b> <b>47</b>	Implementsi API ke UI Fitur 46	0	Wahyu
	Unit Testing Fitur 46	0	Annisa
	Desain Fitur 47	4	Fahmi
	Membuat API Fitur 47	10	Jabbar
	Implementasi UI Fitur 47	8	Wahyu
<b>FITUR</b> <b>48</b>	Implementsi API ke UI Fitur 47	8	Wahyu
	Unit Testing Fitur 47	2	Annisa
	Desain Fitur 48	2	Fahmi
	Membuat API Fitur 48	3	Jabbar
	Implementasi UI Fitur 48	2	Wahyu
	Implementsi API ke UI Fitur 48	2	Wahyu
	Unit Testing Fitur 48	1	Annisa

## 4.2 Review Sprint

### 4.2.1 Sprint Pertama

Sprint pertama dimulai tanggal 07 September s.d 21 September 2021 (2 minggu). Total user stories pada sprint ini sebanyak 152 user stories (118 planning awal + 34 backlog tambahan). dibagikan kepada 4 orang oleh scrum master.

Tabel 6. Sprint Planning Pertama

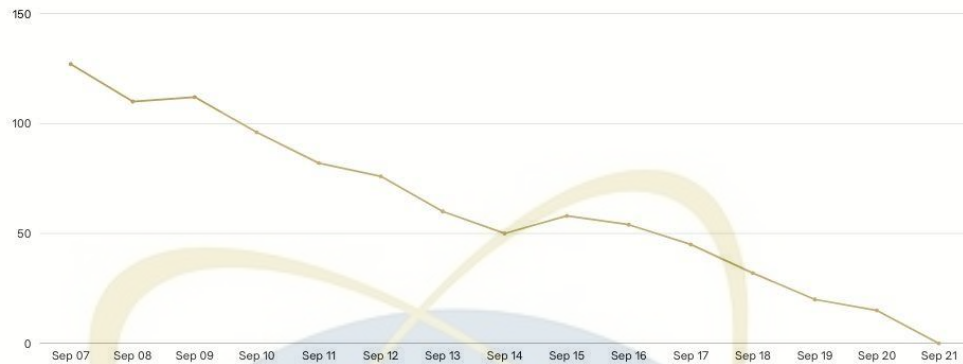
Tasks	Story Points (hours)
-------	----------------------

Desain Fitur 1	6
Membuat Backend API Fitur 1	5
Implementasi UI Fitur 1	6
Desain Fitur 2	2
Membuat API Fitur 2	2
Implementasi UI Fitur 2	4
Desain Fitur 3	3
Membuat API Fitur 3	3
Implementasi UI Fitur 3	3
Desain Fitur 4	3
Membuat API Fitur 4	3
Implementasi UI Fitur 4	3
Desain Fitur 6	2
Membuat API Fitur 6	2
Implementasi UI Fitur 6	2
Desain Fitur 7	3
Membuat API Fitur 7	3
Implementasi UI Fitur 7	3
Desain Fitur 8	4
Membuat API Fitur 8	6
Implementasi UI Fitur 8	8
Desain Fitur 11	3
Membuat API Fitur 11	4
Desain Fitur 12	3
Membuat API Fitur 12	4
Desain Fitur 15	4
Membuat API Fitur 15	6
Implementasi UI Fitur 15	5
Desain Fitur 16	4
Membuat API Fitur 16	5
Implementasi UI Fitur 16	4
<b>Total</b>	<b>118</b>

Tabel 7. Penambahan Backlog Sprint Pertama

Tasks	story points
Desain Fitur 49	2
Membuat API Fitur 49	4
Implementasi UI Fitur 49	10
Desain Fitur 50	3
Membuat API Fitur 50	3
Implementasi UI Fitur 50	3
Desain Fitur 51	5
Membuat API Fitur 51	4
<b>Total</b>	<b>34</b>

## SPRINT 1



Gambar 8. Burndown chart sprint pertama

Tabel 8. Review sprint pertama

Tasks	story points	status
Desain Fitur 1	6	Done
Membuat Backend API Fitur 1	5	Done
Implementasi UI Fitur 1	6	Done
Desain Fitur 2	2	Done
Membuat API Fitur 2	2	Done
Implementasi UI Fitur 2	4	Done
Desain Fitur 3	3	Done
Membuat API Fitur 3	3	Done
Implementasi UI Fitur 3	3	Done
Desain Fitur 4	3	Done
Membuat API Fitur 4	3	Done
Implementasi UI Fitur 4	3	Done
Desain Fitur 6	2	Done
Membuat API Fitur 6	2	Done
Implementasi UI Fitur 6	2	Done
Desain Fitur 7	3	Done
Membuat API Fitur 7	3	Done
Implementasi UI Fitur 7	3	Done
Desain Fitur 8	4	Done
Membuat API Fitur 8	6	Done
Implementasi UI Fitur 8	8	Done
Desain Fitur 11	3	Done
Membuat API Fitur 11	4	Done
Desain Fitur 12	3	Done
Membuat API Fitur 12	4	Done
Desain Fitur 15	4	Done
Membuat API Fitur 15	6	Done



Implementasi UI Fitur 15	5	Done
Desain Fitur 16	4	Done
Membuat API Fitur 16	5	Done
Implementasi UI Fitur 16	4	Done
Desain Fitur 49	2	Done
Membuat API Fitur 49	4	Done
Implementasi UI Fitur 49	10	Done
Desain Fitur 50	3	Done
Membuat API Fitur 50	3	Done
Implementasi UI Fitur 50	3	Done
Desain Fitur 51	5	Done
Membuat API Fitur 51	4	Done
<b>Total Increment</b>	<b>152</b>	

### Sprint Review & Restrospective sprint Pertama

Pada sprint pertama telah selesai beberapa fitur priority berupa desain, implementasi UI dan membuat Backend API. Tampilan dan backend api langsung di deploy secara online melalui tools *vercel* dan *heroku*. Untuk sprint selanjutnya bisa memberikan task implementasi API ke UI pada API-API yang telah dibuat, sembari melanjutkan task kepada tim developer desain.

Sprint retrospective pada sprint pertama terlihat sangat baik, walaupun ada penambahan task di tengah sprint, developer tetap dapat menyelesaikan semua task sampai akhir sprint 1 tanpa terisita. Bisa dilihat total increment sebanyak 152 story points, menandakan performa developer di tim sangat baik.

#### 4.2.2 Sprint Kedua

Sprint pertama Kedua tanggal 22 September s.d 06 Oktober 2021 (2 minggu). Total user stories pada sprint ini sebanyak 145 user stories (114 planning awal + 31 backlog tambahan). dibagikan kepada 4 orang oleh scrum master.

Tabel 9. Sprint Planning Kedua

Tasks	Story Points (hours)
-------	----------------------

Implementasi API ke UI Fitur 1	4
Unit Testing Fitur 1	2
Implementasi API ke UI Fitur 2	2
Unit Testing Fitur 2	2
Implementasi API ke UI Fitur 3	3
Unit Testing Fitur 3	2
Implementasi API ke UI Fitur 4	3
Unit Testing Fitur 4	2
Implementasi API ke UI Fitur 6	3
Implementasi API ke UI Fitur 7	3
Implementasi API ke UI Fitur 8	4
Implementasi UI Fitur 11	7
Implementasi UI Fitur 12	5
Implementasi API ke UI Fitur 15	3
Unit Testing Fitur 15	2
Implementasi API ke UI Fitur 16	3
Unit Testing Fitur 16	2
Desain Fitur 17	4
Membuat API Fitur 17	8
Implementasi UI Fitur 17	6
Desain Fitur 20	5
Membuat API Fitur 20	5
Implementasi UI Fitur 20	4
Desain Fitur 21	3
Membuat API Fitur 21	4
Implementasi UI Fitur 21	3
Desain Fitur 22	3
Membuat API Fitur 22	4
Implementasi UI Fitur 22	3
Desain Fitur 23	3
Membuat API Fitur 23	4
Implementasi UI Fitur 23	3
<b>Total</b>	<b>114</b>

Tabel 10. Penambahan Backlog Sprint Kedua

Tasks	story points
Implementasi API ke UI Fitur 49	5
Unit Testing Fitur 49	2
Implementasi API ke UI Fitur 50	3
Implementasi UI Fitur 51	5
Implementasi API ke UI Fitur 51	5
Merubah API fitur 52	3
Implementasi API ke UI Fitur 52	2
Merubah API fitur 53	3
Implementasi API ke UI Fitur 53	3
<b>Total</b>	<b>31</b>

## SPRINT 2



Gambar 9. Burndown chart sprint kedua

Tabel 11. Review Sprint Kedua

Tasks	story points	status
Implementasi API ke UI Fitur 1	4	Done
Unit Testing Fitur 1	2	Done
Implementasi API ke UI Fitur 2	2	Done
Unit Testing Fitur 2	2	Done
Implementasi API ke UI Fitur 3	3	Done
Unit Testing Fitur 3	2	Done
Implementasi API ke UI Fitur 4	3	Done
Unit Testing Fitur 4	2	Done
Implementasi API ke UI Fitur 6	3	Done
Implementasi API ke UI Fitur 7	3	Done
Implementasi API ke UI Fitur 8	4	Done
Implementasi UI Fitur 11	7	Done
Implementasi UI Fitur 12	5	Done
Implementasi API ke UI Fitur 15	3	Done
Unit Testing Fitur 15	2	Done
Implementasi API ke UI Fitur 16	3	Done
Unit Testing Fitur 16	2	Done
Desain Fitur 17	4	Done
Membuat API Fitur 17	8	Not Done
Implementasi UI Fitur 17	6	Done
Desain Fitur 20	5	Done
Membuat API Fitur 20	5	Not Done
Implementasi UI Fitur 20	4	Done
Desain Fitur 21	3	Done
Membuat API Fitur 21	4	Not Done
Implementasi UI Fitur 21	3	Done
Desain Fitur 22	3	Done

Membuat API Fitur 22	4	Done
Implementasi UI Fitur 22	3	Done
Desain Fitur 23	3	Done
Membuat API Fitur 23	4	Done
Implementasi UI Fitur 23	3	Done
Implementasi API ke UI Fitur 49	5	Done
Unit Testing Fitur 49	2	Done
Implementasi API ke UI Fitur 50	3	Done
Implementasi UI Fitur 51	5	Done
Implementasi API ke UI Fitur 51	5	Done
Merubah API fitur 52	3	Done
Implementasi API ke UI Fitur 52	2	Done
Merubah API fitur 53	3	Done
Implementasi API ke UI Fitur 53	3	Done
<b>Total Increment</b>		<b>128</b>

### Sprint Review & Restrospective sprint Kedua

Pada sprint Kedua telah selesai melanjutkan beberapa fitur secara lengkap hingga implementasi API dan Unit Testing. Kemudian beberapa fitur priority berupa desain, implementasi UI dan membuat Backend API pada fitur lain juga telah dikerjakan. Product owner juga sudah dapat melihat beberapa progress live lewat <http://puskarier.vercel.app>.

Sprint retrospective pada sprint pertama terlihat Cukup baik, tim bekerja dengan maksimal dan hanya menyisakan sedikit saja task yang tidak dapat ditangani. Bisa dilihat total increment sebanyak 128 story points dari planning 145, menandakan performa developer di tim sangat baik walaupun terdapat beberapa task yang tidak terselesaikan dan dimundurkan ke sprint berikutnya.

#### 4.2.3 Sprint Ketiga

Sprint ketiga tanggal 07 Oktober s.d 21 Oktober 2021 (2 minggu). Total story points pada sprint ini sebanyak 154 story points (132 planning awal + 22 backlog tambahan). Dibagikan kepada 4 orang oleh scrum master.

Tabel 12. Sprint Planning Ketiga

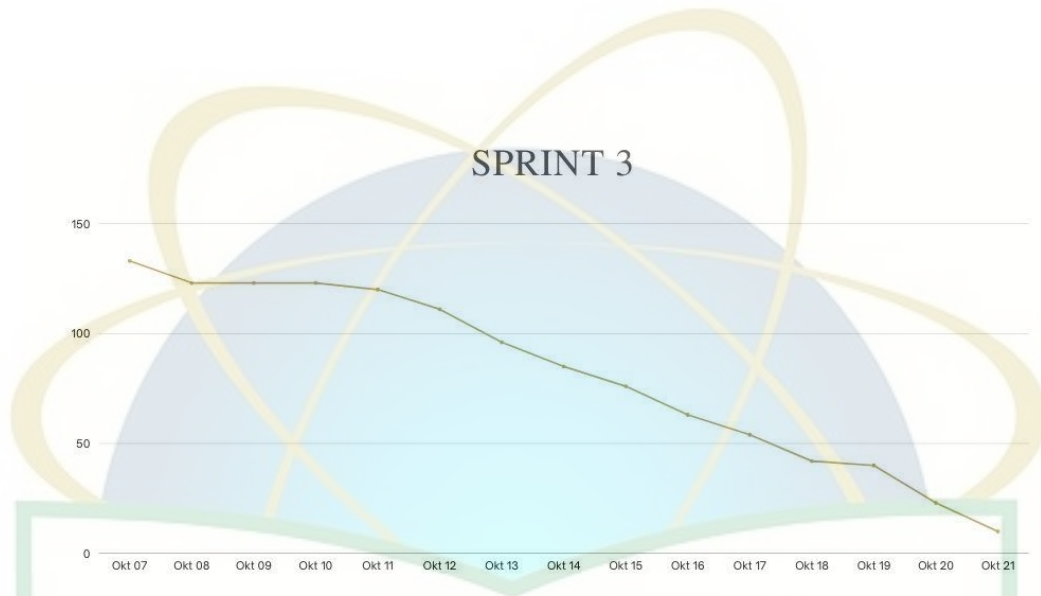
Tasks	Story Points (hours)
-------	----------------------

Membuat API Fitur 17	8
Membuat API Fitur 20	5
Membuat API Fitur 21	4
Unit Testing Fitur 6	1
Unit Testing Fitur 7	2
Unit Testing Fitur 8	2
Implementasi API ke UI Fitur 11	4
Unit Testing Fitur 11	2
Implementasi API ke UI Fitur 17	4
Unit Testing Fitur 17	2
Implementasi API ke UI Fitur 20	4
Unit Testing Fitur 20	2
Desain Fitur 29	4
Membuat API Fitur 29	4
Implementasi UI Fitur 29	3
Desain Fitur 30	4
Membuat API Fitur 30	4
Implementasi UI Fitur 30	3
Desain Fitur 33	4
Membuat API Fitur 33	6
Implementasi UI Fitur 33	5
Desain Fitur 38	2
Membuat API Fitur 38	3
Implementasi UI Fitur 38	2
Desain Fitur 39	2
Membuat API Fitur 39	3
Implementasi UI Fitur 39	2
Desain Fitur 40	4
Unit Testing Fitur 50	2
Unit Testing Fitur 51	1
Implementasi API ke UI Fitur 12	6
Unit Testing Fitur 12	2
Implementasi API ke UI Fitur 21	3
Unit Testing Fitur 21	2
Implementasi API ke UI Fitur 22	3
Unit Testing Fitur 22	2
Implementasi API ke UI Fitur 23	3
Unit Testing Fitur 23	2
Desain Fitur 41	3
Desain Fitur 42	2
Desain Fitur 5	2
Desain Fitur 13	2
Desain Fitur 14	2
<b>Total</b>	<b>132</b>

Tabel 13. Penambahan Backlog Sprint Ketiga

Tasks	story points
Merubah API fitur 54	3

Implementasi API ke UI Fitur 54	3
Unit Testing Fitur 54	1
Merubah API fitur 55	3
Implementasi API ke UI Fitur 55	3
Desain Fitur 56	3
Membuat API Fitur 56	3
Implementasi UI Fitur 56	3
<b>Total</b>	<b>22</b>



Gambar 10. Burndown chart sprint ketiga

Tabel 14. Review sprint ketiga

Tasks	story points	status
Membuat API Fitur 17	8	Done
Membuat API Fitur 20	5	Done
Membuat API Fitur 21	4	Done
Unit Testing Fitur 6	1	Done
Unit Testing Fitur 7	2	Done
Unit Testing Fitur 8	2	Done
Implementasi API ke UI Fitur 11	4	Done
Unit Testing Fitur 11	2	Done
Implementasi API ke UI Fitur 17	4	Done
Unit Testing Fitur 17	2	Done
Implementasi API ke UI Fitur 20	4	Done
Unit Testing Fitur 20	2	Done
Desain Fitur 29	4	Done
Membuat API Fitur 29	4	Done
Implementasi UI Fitur 29	3	Done
Desain Fitur 30	4	Done



Membuat API Fitur 30	4	Done
Implementasi UI Fitur 30	3	Done
Desain Fitur 33	4	Done
Membuat API Fitur 33	6	Done
Implementasi UI Fitur 33	5	Done
Desain Fitur 38	2	Done
Membuat API Fitur 38	3	Not Done
Implementasi UI Fitur 38	2	Not Done
Desain Fitur 39	2	Done
Membuat API Fitur 39	3	Not Done
Implementasi UI Fitur 39	2	Not Done
Desain Fitur 40	4	Done
Unit Testing Fitur 50	2	Done
Unit Testing Fitur 51	1	Done
Implementasi API ke UI Fitur 12	6	Done
Unit Testing Fitur 12	2	Done
Implementasi API ke UI Fitur 21	3	Done
Unit Testing Fitur 21	2	Done
Implementasi API ke UI Fitur 22	3	Done
Unit Testing Fitur 22	2	Done
Implementasi API ke UI Fitur 23	3	Done
Unit Testing Fitur 23	2	Done
Desain Fitur 41	3	Done
Desain Fitur 42	2	Done
Desain Fitur 5	2	Done
Desain Fitur 13	2	Done
Desain Fitur 14	2	Done
Merubah API fitur 54	3	Done
Implementasi API ke UI Fitur 54	3	Done
Unit Testing Fitur 54	1	Done
Merubah API fitur 55	3	Done
Implementasi API ke UI Fitur 55	3	Done
Desain Fitur 56	3	Done
Membuat API Fitur 56	3	Done
Implementasi UI Fitur 56	3	Done
<b>Total Increment</b>	<b>144</b>	

### Sprint Review & Restrospective sprint Ketiga

Pada sprint ketiga telah terselesaikan hampir semua backlog yang prioritynya *high*. Halaman admin juga bisa diakses secara live lewat <https://admin-puskarier.vercel.app> untuk memudahkan product owner mengevaluasi sambil berjalannya pengembangan.

Sprint retrospective pada sprint ketiga masih terlihat Cukup baik. Bisa dilihat total increment sebanyak 144 story points dari planning 154,

menandakan performa developer di tim sangat baik walaupun terdapat beberapa task yang tidak terselesaikan dan dimundurkan ke sprint berikutnya.

#### 4.2.4 Sprint Keempat

Sprint keempat tanggal 22 Oktober s.d 05 November 2021 (2 minggu). Total story points pada sprint ini sebanyak 133 story points (110 planning awal + 23 backlog tambahan). Dibagikan kepada 4 orang oleh scrum master.

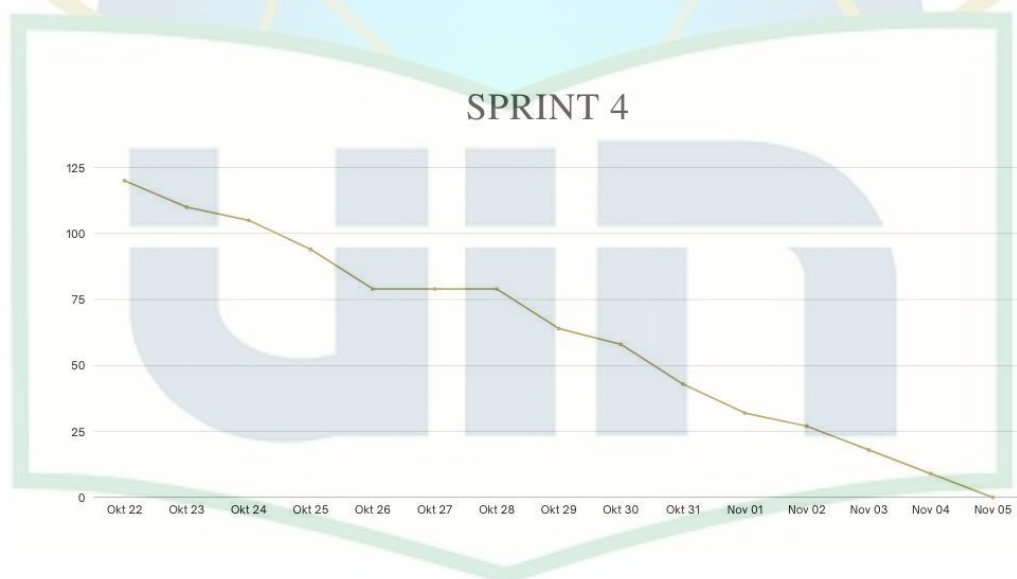
Tabel 15. Sprint Planning Keempat

Tasks	Story Points (hours)
Membuat API Fitur 38	3
Implementasi UI Fitur 38	2
Membuat API Fitur 39	3
Implementasi UI Fitur 39	2
Membuat API Fitur 5	3
Implementasi UI Fitur 5	2
Membuat API Fitur 13	3
Implementasi UI Fitur 13	2
Membuat API Fitur 14	4
Implementasi UI Fitur 14	3
Desain Fitur 25	2
Membuat API Fitur 25	3
Implementasi UI Fitur 25	2
Desain Fitur 26	2
Membuat API Fitur 26	3
Implementasi UI Fitur 26	2
Desain Fitur 27	2
Membuat API Fitur 27	3
Implementasi UI Fitur 27	2
Desain Fitur 28	2
Membuat API Fitur 28	4
Implementasi UI Fitur 28	2
Implementasi API ke UI Fitur 29	3
Unit Testing Fitur 29	2
Implementasi API ke UI Fitur 30	3
Unit Testing Fitur 30	2
Implementasi API ke UI Fitur 33	3
Unit Testing Fitur 33	2
Implementasi API ke UI Fitur 38	2
Unit Testing Fitur 38	1
Implementasi API ke UI Fitur 39	2

Unit Testing Fitur 39	1
Membuat API Fitur 40	8
Implementasi UI Fitur 40	6
Membuat API Fitur 41	6
Implementasi UI Fitur 41	5
Membuat API Fitur 42	4
Implementasi UI Fitur 42	4
<b>Total</b>	<b>110</b>

Tabel 16. Penambahan Backlog Sprint Keempat

Tasks	story points
Implementasi API ke UI Fitur 56	3
Unit Testing Fitur 56	2
Desain Fitur 57	2
Implementasi UI Fitur 57	3
Desain Fitur 57	2
Implementasi UI Fitur 57	3
Perbaikan API Fitur 59	4
Perbaikan API Fitur 60	4
<b>Total</b>	<b>23</b>



Gambar 11. Burndown chart sprint keempat

Tabel 17. Review sprint keempat

Tasks	story points	status
Membuat API Fitur 38	3	Done
Implementasi UI Fitur 38	2	Done
Membuat API Fitur 39	3	Done

Implementasi UI Fitur 39	2	Done
Membuat API Fitur 5	3	Done
Implementasi UI Fitur 5	2	Done
Membuat API Fitur 13	3	Done
Implementasi UI Fitur 13	2	Done
Membuat API Fitur 14	4	Done
Implementasi UI Fitur 14	3	Done
Desain Fitur 25	2	Done
Membuat API Fitur 25	3	Done
Implementasi UI Fitur 25	2	Done
Desain Fitur 26	2	Done
Membuat API Fitur 26	3	Done
Implementasi UI Fitur 26	2	Done
Desain Fitur 27	2	Done
Membuat API Fitur 27	3	Done
Implementasi UI Fitur 27	2	Done
Desain Fitur 28	2	Done
Membuat API Fitur 28	4	Done
Implementasi UI Fitur 28	2	Done
Implementasi API ke UI Fitur 29	3	Done
Unit Testing Fitur 29	2	Done
Implementasi API ke UI Fitur 30	3	Done
Unit Testing Fitur 30	2	Done
Implementasi API ke UI Fitur 33	3	Done
Unit Testing Fitur 33	2	Done
Implementasi API ke UI Fitur 38	2	Done
Unit Testing Fitur 38	1	Done
Implementasi API ke UI Fitur 39	2	Done
Unit Testing Fitur 39	1	Done
Membuat API Fitur 40	8	Done
Implementasi UI Fitur 40	6	Done
Membuat API Fitur 41	6	Done
Implementasi UI Fitur 41	5	Done
Membuat API Fitur 42	4	Done
Implementasi UI Fitur 42	4	Done
Implementasi API ke UI Fitur 56	3	Done
Unit Testing Fitur 56	2	Done
Desain Fitur 57	2	Done
Implementasi UI Fitur 57	3	Done
Desain Fitur 57	2	Done
Implementasi UI Fitur 57	3	Done
Perbaikan API Fitur 59	4	Done
Perbaikan API Fitur 60	4	Done
<b>Total Increment</b>	<b>133</b>	

### Sprint Review & Restrospective sprint Keempat

Pada Sprint Keempat sudah berhasil menyelesaikan fitur-fitur yang High Priority, beberapa fitur juga sudah di unit testing sehingga dapat langsung di tes oleh Product owner. Beberapa fitur yang belum terselesaikan di sprint sebelumnya di masukkan kedalam sprint ini.

Sprint retrospective pada sprint keempat terlihat Cukup baik, walaupun ada penambahan task ditengah sprint, developer tetap dapat menyelesaikan semua task sampai akhir sprint 4 tanpa terisira. Bisa dilihat total increment sebanyak 133 dari 133 total story point sprint 4, menandakan performa developer di tim sangat baik.

#### 4.2.5 Sprint Kelima

Sprint kelima tanggal 06 November s.d 20 November 2021 (2 minggu). Total story points pada sprint ini sebanyak 141 story points (121 planning awal + 20 backlog tambahan). Dibagikan kepada 4 orang oleh scrum master.

Tabel 18. Sprint Planning Kelima

Tasks	Story Points (hours)
Implementasi API ke UI Fitur 5	2
Unit Testing Fitur 5	1
Desain Fitur 9	2
Desain Fitur 10	2
Implementasi API ke UI Fitur 25	2
Unit Testing Fitur 25	1
Implementasi API ke UI Fitur 26	2
Unit Testing Fitur 26	1
Implementasi API ke UI Fitur 27	2
Unit Testing Fitur 27	1
Implementasi API ke UI Fitur 28	2
Unit Testing Fitur 28	2
Desain Fitur 31	2
Membuat API Fitur 31	3
Implementasi UI Fitur 31	2
Desain Fitur 32	2
Membuat API Fitur 32	3
Implementasi UI Fitur 32	2
Desain Fitur 34	2
Membuat API Fitur 34	3
Implementasi UI Fitur 34	2

Desain Fitur 35	2
Membuat API Fitur 35	3
Implementasi UI Fitur 35	2
Desain Fitur 36	2
Membuat API Fitur 36	3
Implementasi UI Fitur 36	4
Desain Fitur 37	2
Membuat API Fitur 37	5
Implementasi UI Fitur 37	4
Implementasi API ke UI Fitur 40	6
Unit Testing Fitur 40	2
Implementasi API ke UI Fitur 41	4
Unit Testing Fitur 41	2
Implementasi API ke UI Fitur 42	4
Unit Testing Fitur 42	2
Desain Fitur 43	2
Desain Fitur 44	2
Desain Fitur 47	4
Membuat API Fitur 47	10
Implementasi UI Fitur 47	8
Desain Fitur 48	2
Membuat API Fitur 48	3
Implementasi UI Fitur 48	2
<b>Total</b>	<b>121</b>

Tabel 19. Penambahan Backlog Sprint Kelima

Tasks	story points
Unit Testing Fitur 58	1
Unit Testing Fitur 59	1
Unit Testing Fitur 60	1
Perbaikan API Fitur 61	2
Implementasi UI ke API Fitur 61	2
Perbaikan API Fitur 62	4
Implementasi UI ke API Fitur 62	3
Perbaikan UI Fitur 63	3
Perbaikan API Fitur 63	3
<b>Total</b>	<b>20</b>



## SPRINT 5



Gambar 12. Burndown chart sprint kelima

Tabel 20. Review sprint kelima

Tasks	story points	status
Implementasi API ke UI Fitur 5	2	Done
Unit Testing Fitur 5	1	Done
Desain Fitur 9	2	Done
Desain Fitur 10	2	Done
Implementasi API ke UI Fitur 25	2	Done
Unit Testing Fitur 25	1	Done
Implementasi API ke UI Fitur 26	2	Done
Unit Testing Fitur 26	1	Done
Implementasi API ke UI Fitur 27	2	Done
Unit Testing Fitur 27	1	Done
Implementasi API ke UI Fitur 28	2	Done
Unit Testing Fitur 28	2	Done
Desain Fitur 31	2	Done
Membuat API Fitur 31	3	Done
Implementasi UI Fitur 31	2	Done
Desain Fitur 32	2	Done
Membuat API Fitur 32	3	Done
Implementasi UI Fitur 32	2	Done
Desain Fitur 34	2	Done
Membuat API Fitur 34	3	Done
Implementasi UI Fitur 34	2	Done
Desain Fitur 35	2	Done
Membuat API Fitur 35	3	Done
Implementasi UI Fitur 35	2	Done
Desain Fitur 36	2	Done
Membuat API Fitur 36	3	Not Done
Implementasi UI Fitur 36	4	Not Done

Desain Fitur 37	2	Done
Membuat API Fitur 37	5	Not Done
Implementasi UI Fitur 37	4	Not Done
Implementasi API ke UI Fitur 40	6	Done
Unit Testing Fitur 40	2	Done
Implementasi API ke UI Fitur 41	4	Done
Unit Testing Fitur 41	2	Done
Implementasi API ke UI Fitur 42	4	Done
Unit Testing Fitur 42	2	Done
Desain Fitur 43	2	Done
Desain Fitur 44	2	Done
Desain Fitur 47	4	Done
Membuat API Fitur 47	10	Done
Implementasi UI Fitur 47	8	Done
Desain Fitur 48	2	Done
Membuat API Fitur 48	3	Done
Implementasi UI Fitur 48	2	Done
Unit Testing Fitur 58	1	Done
Unit Testing Fitur 59	1	Done
Unit Testing Fitur 60	1	Done
Perbaikan API Fitur 61	2	Done
Implementasi UI ke API Fitur 61	2	Done
Perbaikan API Fitur 62	4	Done
Implementasi UI ke API Fitur 62	3	Done
Perbaikan UI Fitur 63	3	Done
Perbaikan API Fitur 63	3	Done
<b>Total Increment</b>	<b>125</b>	

### **Sprint Review & Restrospective sprint Keempat**

Pada Sprint Kelima, fitur aplikasi hampir sepenuhnya terselesaikan, beberapa fitur sudah di unit testing dan berhasil dengan optimal. Terdapat juga tambahan-tambahan yang diberikan oleh product owner yang kemudian oleh scrum master dimasukkan kedalam perubahan product backlog.

Sprint retrospective pada sprint kelima terlihat Cukup baik, walaupun ada penambahan task ditengah sprint, developer tetap dapat menyelesaikan hampir semua backlog. Bisa dilihat total increment sebanyak 125 dari 141 total story point sprint 5, menandakan performa developer di tim masih sangat baik.

#### 4.2.6 Sprint Keenam

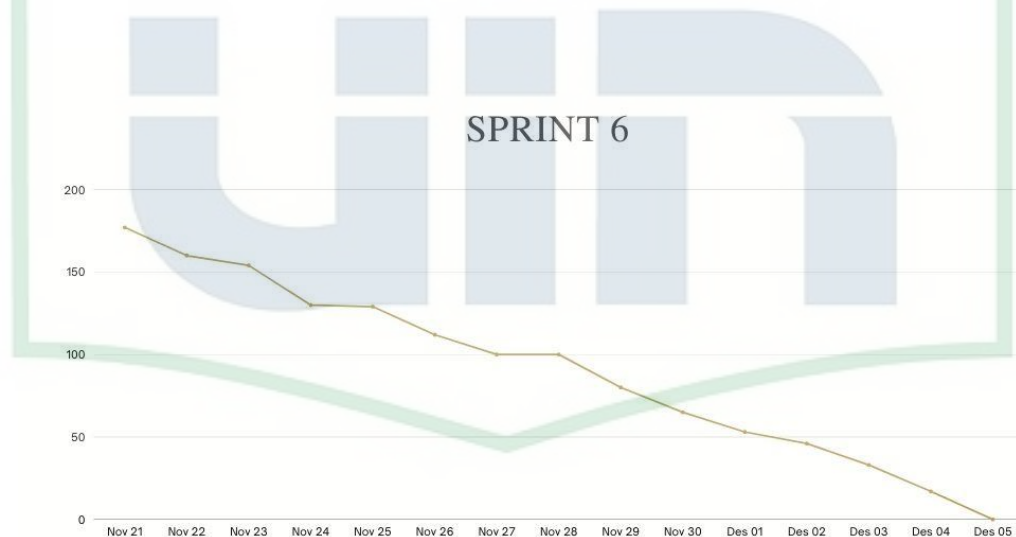
Sprint keenam tanggal 21 November s.d 05 Desember 2021 (2 minggu). Total story points pada sprint ini sebanyak 177 story points (113 planning awal + 64 backlog tambahan). Dibagikan kepada 4 orang oleh scrum master.

Tabel 21. Sprint Planning Keenam

Tasks	Story Points (hours)
Membuat API Fitur 36	3
Implementasi UI Fitur 36	4
Membuat API Fitur 37	5
Implementasi UI Fitur 37	4
Membuat API Fitur 9	6
Implementasi UI Fitur 9	6
Implementasi API ke UI Fitur 9	4
Unit Testing Fitur 9	2
Membuat API Fitur 10	6
Implementasi UI Fitur 10	6
Implementasi API ke UI Fitur 10	4
Unit Testing Fitur 10	2
Implementasi API ke UI Fitur 13	2
Unit Testing Fitur 13	1
Implementasi API ke UI Fitur 14	2
Unit Testing Fitur 14	1
Implementasi API ke UI Fitur 34	2
Unit Testing Fitur 34	1
Implementasi API ke UI Fitur 35	2
Unit Testing Fitur 35	1
Implementasi API ke UI Fitur 36	2
Unit Testing Fitur 36	1
Implementasi API ke UI Fitur 37	3
Unit Testing Fitur 37	2
Membuat API Fitur 43	5
Implementasi UI Fitur 43	4
Implementasi API ke UI Fitur 43	4
Unit Testing Fitur 43	1
Membuat API Fitur 44	5
Implementasi UI Fitur 44	4
Implementasi API ke UI Fitur 44	4
Unit Testing Fitur 44	1
Implementasi API ke UI Fitur 47	8
Unit Testing Fitur 47	2
Implementasi API ke UI Fitur 48	2
Unit Testing Fitur 48	1
<b>Total</b>	<b>113</b>

Tabel 22. Penambahan Backlog Sprint Keenam

Tasks	story points
Unit Testing Fitur 61	1
Unit Testing Fitur 62	1
Perbaikan Implementasi UI Fitur 69	4
Unit Testing Fitur 69	1
Perbaikan API Fitur 70	4
Implementasi API Ke UI Fitur 70	4
Unit Testing Fitur 70	2
Perbaikan Implementasi UI Fitur 71	4
Unit Testing Fitur 71	1
Perbaikan Implementasi UI Fitur 72	4
Unit Testing Fitur 72	1
Perbaikan Implementasi UI Fitur 73	4
Unit Testing Fitur 73	1
Perbaikan Implementasi UI Fitur 74	4
Unit Testing Fitur 74	1
Perbaikan API Fitur 70	6
Implementasi API Ke UI Fitur 70	4
Unit Testing Fitur 70	2
Perbaikan API Fitur 70	6
Implementasi API Ke UI Fitur 70	7
Unit Testing Fitur 70	2
<b>Total</b>	<b>64</b>



Gambar 13. Burndown chart sprint keenam

Tabel 23. Review sprint keenam

Tasks	story points	status
Membuat API Fitur 36	3	Done
Implementasi UI Fitur 36	4	Done
Membuat API Fitur 37	5	Done
Implementasi UI Fitur 37	4	Done
Membuat API Fitur 9	6	Done
Implementasi UI Fitur 9	6	Done
Implementasi API ke UI Fitur 9	4	Done
Unit Testing Fitur 9	2	Done
Membuat API Fitur 10	6	Done
Implementasi UI Fitur 10	6	Done
Implementasi API ke UI Fitur 10	4	Done
Unit Testing Fitur 10	2	Done
Implementasi API ke UI Fitur 13	2	Done
Unit Testing Fitur 13	1	Done
Implementasi API ke UI Fitur 14	2	Done
Unit Testing Fitur 14	1	Done
Implementasi API ke UI Fitur 34	2	Done
Unit Testing Fitur 34	1	Done
Implementasi API ke UI Fitur 35	2	Done
Unit Testing Fitur 35	1	Done
Implementasi API ke UI Fitur 36	2	Done
Unit Testing Fitur 36	1	Done
Implementasi API ke UI Fitur 37	3	Done
Unit Testing Fitur 37	2	Done
Membuat API Fitur 43	5	Done
Implementasi UI Fitur 43	4	Done
Implementasi API ke UI Fitur 43	4	Done
Unit Testing Fitur 43	1	Done
Membuat API Fitur 44	5	Done
Implementasi UI Fitur 44	4	Done
Implementasi API ke UI Fitur 44	4	Done
Unit Testing Fitur 44	1	Done
Implementasi API ke UI Fitur 47	8	Done
Unit Testing Fitur 47	2	Done
Implementasi API ke UI Fitur 48	2	Done
Unit Testing Fitur 48	1	Done
Unit Testing Fitur 61	1	Done
Unit Testing Fitur 62	1	Done
Perbaikan Implementasi UI Fitur 69	4	Done
Unit Testing Fitur 69	1	Done
Perbaikan API Fitur 70	4	Done
Implementasi API Ke UI Fitur 70	4	Done
Unit Testing Fitur 70	2	Done
Perbaikan Implementasi UI Fitur 71	4	Done
Unit Testing Fitur 71	1	Done
Perbaikan Implementasi UI Fitur 72	4	Done
Unit Testing Fitur 72	1	Done

Perbaikan Implementasi UI Fitur 73	4	Done
Unit Testing Fitur 73	1	Done
Perbaikan Implementasi UI Fitur 74	4	Done
Unit Testing Fitur 74	1	Done
Perbaikan API Fitur 70	6	Done
Implementasi API Ke UI Fitur 70	4	Done
Unit Testing Fitur 70	2	Done
Perbaikan API Fitur 70	6	Done
Implementasi API Ke UI Fitur 70	7	Done
Unit Testing Fitur 70	2	Done
<b>Total Increment</b>	<b>177</b>	

### **Sprint Review & Restrospective sprint Keenam**

Pada Sprint keenam sudah berhasil semua fitur pada list backlog, semua fitur sudah di testing dan siap dites akhir kepada product owner menggunakan unit test. Semua data dummy juga sudah disiapkan guna kelancaran saat tes langsung oleh product owner.

Sprint retrospective pada sprint keenam bisa dikatakan sangat baik, walaupun terdapat banyak backlog yang tersisa, tim pengembang dapat memaksimalkan waktu yang tersisa untuk menyelesaikan semua backlog yang tersisa, total 177 story points terselesaikan pada sprint keenam ini, bisa dikatakan performa terbaik tim developer terbukti saat sprint ini.

### **4.3 Perubahan Product Backlog**

Menurut survey dari Ahmed Mubark Alsalemi yang berjudul “A Survey on Product Backlog Change Management and Requirement Traceability in Agile (Serum)” terdapat 4 faktor alasan terkuat dalam perubahan product backlog diantaranya, penyesuaian requirements, penambahan requirements, perbaikan bug / cacat, dan pengurangan scope (Alsalemi & Yeoh, 2016).

Penulis mengambil 4 alasan tersebut untuk membedakan alasan-alasan yang terjadi pada setiap perubahan product backlog. berikut paparan perubahan product backlog yang didapat selama pengembangan sistem informasi Puskarier.

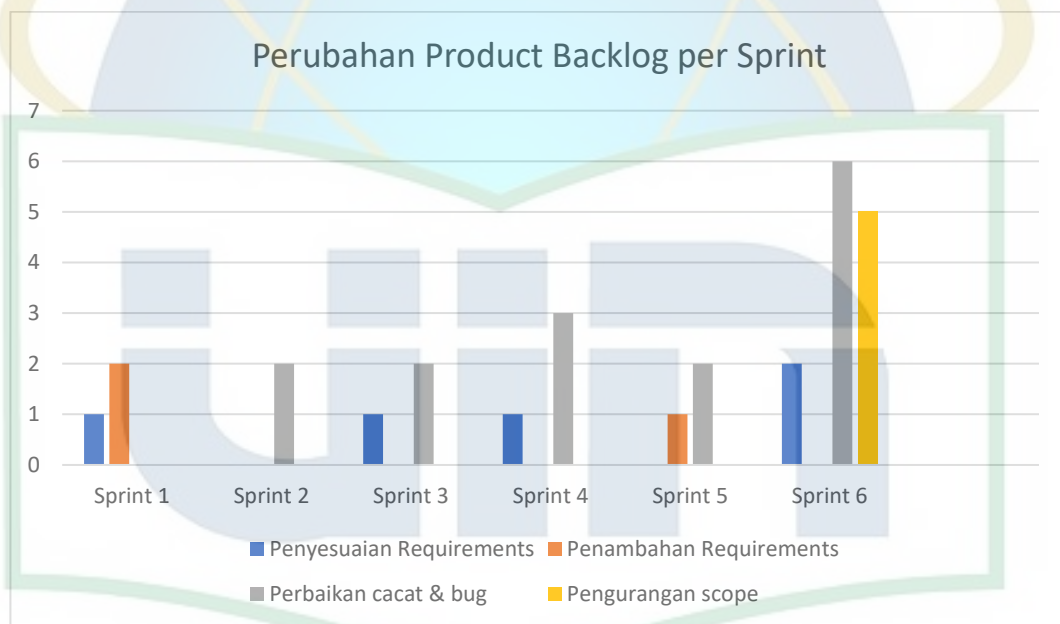


Tabel 24. Perubahan product backlog dalam pengembangan puskarier

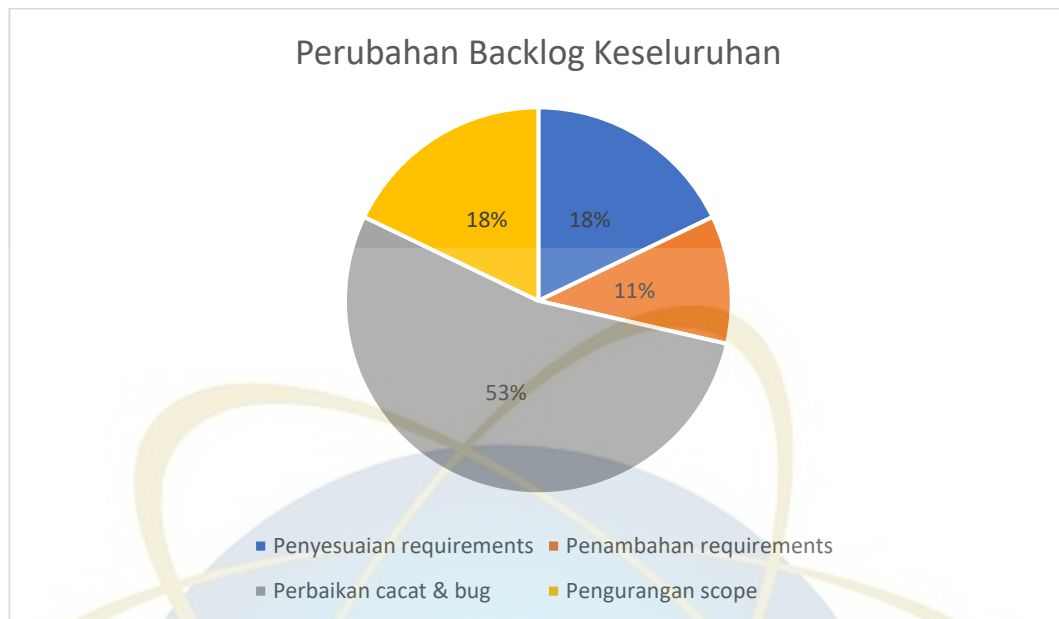
<i>ID Fitur</i>	<i>Sebagai</i>	<i>User Stories</i>	<i>Story Point</i>	<i>Priority</i>	<i>Alasan</i>
49	Admin	Membuat halaman login terpisah untuk admin	23	High	Penyesuaian requirements
50	Pencaker	Menambah Fitur Forgot Password	14	Medium	Penambahan requirements
51	Pencaker	Menambah Fitur Halaman Home	20	Medium	Penambahan requirements
52	Pencaker	Perbaikan bug mengirim email verifikasi	5	Medium	Perbaikan cacat
53	Pencaker	Perbaikan Flow Reset Password	6	Medium	Perbaikan cacat
54	Pencaker	Perbaikan Bug tidak bisa memfilter lowongan	7	Medium	Perbaikan cacat
55	Pencaker	Memberikan alert jika ingin keluar dari halaman mengisi tracer study	6	Medium	Perbaikan cacat
56	Admin	Perubahan jawaban tracer study sehingga memiliki tipe	14	High	Penyesuaian Requirements
57	Pencaker	Penyesuaian Tampilan registrasi, penambahan syarat ketentuan dan persetujuan pencari kerja	6	Medium	Penyesuaian Requirements
58	Pencaker	Perbaikan Tampilan UI Halaman Ubah Profile Pencari Kerja	6	Medium	Perbaikan cacat
59	Admin	Perbaikan API update data pencari kerja	4	Medium	Perbaikan cacat
60	Pencaker	Perbaikan Bug Notifikasi pencari kerja	4	Medium	Perbaikan cacat
61	Alumni	Perbaikan bug alumni tidak dapat menyimpan data saat mendadak keluar	5	Medium	Perbaikan cacat
62	Pencaker	Perbaikan Bug tidak bisa melakukan login pada akun yang didaftarkan admin	8	Medium	Perbaikan cacat
63	Admin	Menambah Halaman Prolog Login, Deskripsi, dan Prolog Logout pada pengisian tracer study	10	Medium	Penambahan requirements
64	Admin	Penghapusan fitur 24 aktivitas akun	0	Medium	Pengurangan Scope
65	Perusahaan	Penghapusan fitur 18 Kontrak Berlangganan	0	Medium	Pengurangan Scope
66	Perusahaan	Penghapusan fitur 19 Riwayat Kontrak Berlangganan	0	Medium	Pengurangan Scope
67	Admin	Penghapusan fitur 45 Mengelola data paket langganan	0	Medium	Pengurangan Scope
68	Admin	Penghapusan fitur 46 Mengelola data pembayaran	0	Medium	Pengurangan Scope
69	Admin	Penambahan button reload pada setiap table untuk mensiasati data yang terbaru	5	Medium	Perbaikan cacat

70	Admin	Perbaikan jawaban tracer study bertipe "Grid Column Answer" tidak tersimpan	10	Medium	Perbaikan cacat
71	Pencaker	Perbaikan Bug dan Error pada Alert Toast	5	Medium	Perbaikan cacat
72	Pencaker	Perbaikan UI Event Tidak dapat sesuai Figma	5	Medium	Perbaikan cacat
73	Pencaker	Perbaikan bug pada tampilan mobile halaman Home	5	Medium	Perbaikan cacat
74	Admin	Perbaikan bug pada tampilan mobile halaman Admin Dashboard	5	Medium	Perbaikan cacat
75	Pencaker	Perubahan flow status pada sistem event	12	Medium	Penyesuaian Requirements
76	Admin	Update Profile pada admin juga menyediakan update password	15	Medium	Penyesuaian Requirements

#### 4.4 Analisa Perubahan Product Backlog



Gambar 14. Perubahan backlog setiap sprint



Gambar 15. Perubahan product backlog selama pengembangan

Berdasarkan pengamatan implementasi scrum pada pengembangan sistem informasi pusat karir UIN Jakarta, didapatkan bahwa faktor tertinggi pada perubahan product backlog adalah Perbaikan cacat & bug. Untuk faktor lain seperti perubahan dan penambahan requirements serta pengurangan scope tidak terlalu besar impact nya terhadap perubahan product backlog.

Dengan data tersebut dapat disimpulkan bahwa pengembangan scrum dengan requirements yang sudah jelas terlebih lagi sudah pernah terimplementasi sebelumnya menghasilkan product backlog yang mudah untuk didefinisikan diawal saat backlog planning, serta perubahan yang terjadi juga tidak banyak, karena bisa dikatakan perangkat lunak yang dikembangkan bisa dikatakan sudah fix dan tidak fleksibel.

#### 4.5 Hasil Pengujian UAT

Dengan menggunakan *blackbox testing* sebagai tipe pengujian UAT didapat hasil pengujian sebagai berikut:

ID Fitur	Skenario / backlog	Hasil yang diharapkan	sesuai (Ya/Tidak)	validator
1	Login dan registrasi	berhasil membuat halaman login dan registrasi serta fungsionalitasnya	Ya	Puskarier
2	Melakukan pencarian	dapat melakukan pencarian pada halaman sistem informasi	Ya	Puskarier
3	Melihat halaman detail pencari kerja	sebagai user dapat melihat halaman detail pencari kerja	Ya	Puskarier
4	Melihat halaman perusahaan	sebagai user dapat melihat halaman perusahaan	Ya	Puskarier
5	Menandai lowongan kerja favorit	dapat menandai lowongan kerja favorit	Ya	Puskarier
6	Melamar pekerjaan	pencari kerja dapat melamar pekerjaan sendiri	Ya	Puskarier
7	Melihat lowongan kerja yang telah dilamar	pencari kerja dapat melihat lowongan kerja yang telah dilamar	Ya	Puskarier
8	Mengelola profile pencari kerja	dapat mengelola profile pencari kerjanya sendiri	Ya	Puskarier
9	Melakukan assessment	pencari kerja dapat melakukan assesmenet	Ya	Puskarier
10	Mengelola konseling kerja	dapat mengelola konseling kerja	Ya	Puskarier
11	Melihat artikel	dapat melihat artikel yang dibuat oleh admin serta dapat membagikannya	Ya	Puskarier
12	Melihat event	dapat melihat event yang dibuat oleh admin puskarier	Ya	Puskarier
13	Daftar event	dapat mendaftar dan memilih event sendiri	Ya	Puskarier
14	Mengelola event yang didaftar	dapat melihat event yang telah didaftar	Ya	Puskarier

15	Mengelola profil perusahaan	perusahaan dapat mengelola profil perusahaannya	Ya	Puskarier
16	Mengelola lowongan kerja	perusahaan dapat mengelola lowongan kerjanya	Ya	Puskarier
17	Mengelola data pelamar	perusahaan dapat mengelola data pelamarnya	Ya	Puskarier
20	Mengisi data tracer study	alumni dapat mengisi data tracer study berdasarkan kategori	Ya	Puskarier
21	Mengelola akun pencaker	admin dapat mengelola akun pencaker	Ya	Puskarier
22	Mengelola akun perusahaan	admin dapat mengelola akun perusahaan	Ya	Puskarier
23	Mengelola akun admin	admin dapat mengelola akun admin	Ya	Puskarier
25	Mengelola data industri pekerjaan	admin dapat mengelola data master industri pekerjaan	Ya	Puskarier
26	Mengelola data peran pekerjaan	admin dapat mengelola data peran pekerjaan	Ya	Puskarier
27	Mengelola kategori artikel	admin dapat mengelola data master kategori artikel	Ya	Puskarier
28	Mengelola artikel	admin dapat mengelola artikel	Ya	Puskarier
29	Mengelola lowongan kerja	admin dapat mengelola lowongan kerja	Ya	Puskarier
30	Mengelola data pelamar	admin dapat mengelola data pelamar setiap perusahaan yang terdaftar	Ya	Puskarier
31	Mengelola data institusi pendidikan	admin dapat mengelola data institusi pendidikan	Ya	Puskarier
32	Mengelola data bidang studi	admin dapat mengelola data bidang studi	Ya	Puskarier
33	Mengelola profil	admin dapat mengelola profilnya	Ya	Puskarier
34	Mengelola FAQ	admin dapat mengelola data master FAQ	Ya	Puskarier
35	Mengelola kategori event	admin dapat mengelola kategori event	Ya	Puskarier

36	Mengelola event	admin dapat mengelola event	Ya	Puskarier
37	Mengelola pendaftar event	admin dapat mengelola pendaftar pada setiap event	Ya	Puskarier
38	Mengelola data fakultas dan prodi	admin dapat mengelola data master ffakultas dan prodi	Ya	Puskarier
39	Mengelola kategori kuesioner tracer study	admin dapat mengelola kategori kuesioner tracer study	Ya	Puskarier
40	Mengelola kuesioner tracer study	admin dapat mengelola kuesioner tracer study	Ya	Puskarier
41	Mengelola data tracer study	admin dapat mengelola data tracer study	Ya	Puskarier
42	Mengelola database alumni	admin dapat mengelola database alumni	Ya	Puskarier
43	Mengelola data konselor karier	admin dapat mengelola konselor karier	Ya	Puskarier
44	Mengelola data pendaftar konseling kerja	admin dapat mengelola data pendaftar konseling kerja	Ya	Puskarier
47	Melihat overview dan statistik	admin dapat mengelola overview dan statistik	Ya	Puskarier
48	Mengelola deskripsi pusat karier	admin dapat mengelola deskripsi pusat karir	Ya	Puskarier



## BAB V

### KESIMPULAN DAN SARAN

#### 5.1 Kesimpulan

Berdasarkan penelitian dan pembahasan pada bab sebelumnya, maka dapat disimpulkan bahwa scrum berhasil dan cocok diterapkan pada pengembangan perangkat lunak pada sistem informasi pusat karir UIN Jakarta, bisa dilihat dari pengembangan yang selesai tepat waktu dan sesuai dengan timeline yang direncanakan.

Berdasarkan analisa perubahan product backlog yang terjadi, terlihat bahwa perubahan terjadi sangat minim pada pengembangan ini. Faktor paling tinggi terjadinya perubahan product backlog adalah adanya Perbaikan cacat & bug (15 kali), kemudian disusul dengan penyesuaian requirements dan pengurangan scope (5x). Hal ini dapat terjadi dikarenakan pendefinisian user stories yang sudah ada mengacu pada aplikasi yang sudah berjalan sekarang, sehingga perubahan requirements (backlog) minim terjadi. Namun tetap, sesuai dengan konsep *agile* bahwa perubahan *backlog* atau *requirements* dapat dengan mudah ditangani dengan *agile*.

Hasil penelitian yang dilakukan peneliti selaras seperti penelitian yang dilakukan oleh Alsalemi & Yeoh yang menyatakan bahwa *defect fixing* / perbaikan cacat & bug kerap terjadi. Dalam penelitiannya, para pelaku scrum mengatakan bahwa defect fixing adalah alasan yang sangat sering terjadi (29%) pada praktik scrum (Alsalemi & Yeoh, 2016).

#### 5.2 Saran

Setelah dilakukan penelitian ini, penulis menyarankan kepada peneliti selanjutnya sebagai bentuk penyempurnaan, sebagai berikut:

- a. Melakukan penelitian penerapan agile pada pengembangan produk yang flexibel
- b. Melanjutkan pengembangan fitur baru puskarier dengan konsep agile

## DAFTAR PUSTAKA

- Alsalemi, A. M., & Yeoh, E. T. (2016). A survey on product backlog change management and requirement traceability in agile (Scrum). *2015 9th Malaysian Software Engineering Conference, MySEC 2015*, 189–194. <https://doi.org/10.1109/MySEC.2015.7475219>
- Apoorva Srivastava, Sukriti Bhardwaj, S. S. (2017). SCRUM Model for Agile Methodology. *International Conference on Computing, Communication and Automation (ICCCA2017)*.
- Bick, S., Spohrer, K., Hoda, R., Scheerer, A., & Heinzl, A. (2018). Coordination Challenges in Large-Scale Software Development: A Case Study of Planning Misalignment in Hybrid Settings. *IEEE Transactions on Software Engineering*, 44(10), 932–950. <https://doi.org/10.1109/TSE.2017.2730870>
- Boehm, B. (2007). A Spiral model of software development and enhancement. *Software Management, Seventh Edition*, 37–48. <https://doi.org/10.1109/9780470049167.ch2>
- Bolung, M., & Tampangela, H. R. K. (2017). Analisa Penggunaan Metodologi Pengembangan Perangkat Lunak. *Jurnal ELTIKOM*, 1(1), 1–10. <https://doi.org/10.31961/eltikom.v1i1.1>
- Department of the Air Force. (2003). *Condensed GSAM Handbook - Guidelines for Successful Acquisition and Management of Software-Intensive Systems: Weapon Systems, Command and Control Systems, Management Information Systems*. February.
- Gonçalves, L. (2018). Scrum: The methodology to become more agile. *Controlling & Management Review*, 4, 40–42. [www.scrumguides.org](http://www.scrumguides.org).
- Hijazi, H., Khmour, T., & Alarabeyyat, A. (2012). A Review of risk management in different software development methodologies. *International Journal of Computer Applications in Technology*, 45(7), 8–12.
- Hoda, R., Salleh, N., & Grundy, J. (2018). The Rise and Evolution of Agile Software Development. *IEEE Software*, 35(5), 58–63. <https://doi.org/10.1109/MS.2018.290111318>
- <https://karir.uinjkt.ac.id>. (2021). *Pusat Karier UIN Jakarta*. <https://karir.uinjkt.ac.id>
- Kayes, I., Sarker, M., & Chakareski, J. (2016). Product backlog rating: a case study on measuring test quality in scrum. *Innovations in Systems and Software Engineering*, 12(4), 303–317. <https://doi.org/10.1007/s11334-016-0271-0>
- KBBI Daring. (n.d.). <https://kbbi.kemdikbud.go.id/entri/investigasi>
- Khan, N. (2016). *SCRUM Adoption : A Solution to Backlog Problems*. October. <https://www.researchgate.net/publication/309734111>
- Lei, H., Ganjeizadeh, F., Jayachandran, P. K., & Ozcan, P. (2017). A statistical analysis of the effects of Scrum and Kanban on software development projects. *Robotics and Computer-Integrated Manufacturing*, 43, 59–67. <https://doi.org/10.1016/j.rcim.2015.12.001>

- Leung, H. K. N., & Wong, P. W. L. (1997). A study of user acceptance tests. *Software Quality Journal*, 6(2), 137–149. <https://doi.org/10.1023/A:1018503800709>
- Liubchenko, V. (2016). A Review of Agile Practices for Project Management. *International Scientific and Technical Conference on Computer Sciences and Information Technologies*, 168–170. <https://doi.org/10.1109/STC-CSIT.2016.7589897>
- Ma'arif, M. Y., Mohd Satar, N. S., Shahrar, S. M., & Yusof, M. F. H. (2018). The Challenges of Implementing Agile Scrum in Information System's Project. *Jour of Adv Research in Dynamical & Control Systems*, 10(09), 2357–2363. <http://www.jardcs.org/abstract.php?archiveid=5261>
- Mathur, S., & Malik, S. (2010). Advancements in the V-Model. *International Journal of Computer Applications*, 1(12), 30–35. <https://doi.org/10.5120/266-425>
- Meyer, B., & Good, T. (2014). *Agile !*
- Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. (2015). *PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN)*. I(3), 31–36.
- Oktriyana, A. D. (2021). *ANALISIS KEBUTUHAN PERANGKAT LUNAK: STUDI KASUS SISTEM INFORMASI PUSAT KARIER UIN SYARIF HIDAYATULLAH JAKARTA*. UIN Syarif Hidayatullah Jakarta.
- Reddaiah, B. (2016). *Handling Transition Product Backlog with Scrum off Scrum*. 5(1), 123–126. <https://doi.org/10.15693/ijaist/2016.v5i1.123-126>
- Ruseno, N. (2019). Implementasi Scrum pada Pengembangan Aplikasi Sistem Reservasi Online Menggunakan PHP. *Gerbang*, 9(1), 8–15. <http://jurnal.stmik.banisaleh.ac.id/index.php/JIST/article/view/61>
- Sedano, T., Ralph, P., & Peraire, C. (2019). The Product Backlog. *Proceedings - International Conference on Software Engineering, 2019-May (Section IV)*, 200–211. <https://doi.org/10.1109/ICSE.2019.00036>
- Seikola, M. (2010). *The Scrum Product Backlog as a Tool for Steering the Product Development in a Large-Scale Organization*. May, 111.
- Silvax, A., Silva, A., Araújo, T., Willamy, R., Ramos, F., Costa, A., Perkusich, M., & Dilozenzo, E. (2017). Ordering the product backlog in agile software development projects: A systematic literature review. *Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE*, 74–80. <https://doi.org/10.18293/SEKE2017-007>
- Szalvay, V., & Suite, B. R. (2004). *An Introduction to Agile Software Development Figure 1. November*.
- University of Mauritius, Institute of Electrical and Electronics Engineers. Mauritius Subsection, Mauritius Research Council, & Institute of Electrical and Electronics Engineers. (2017). 2017 1st International Conference on Next Generation Computing Applications (NextComp) : 19th-21st July 2017, Mauritius. *2017 1st International Conference on Next Generation Computing Applications (NextComp) : 19th-21st July 2017, Mauritius*, 5.

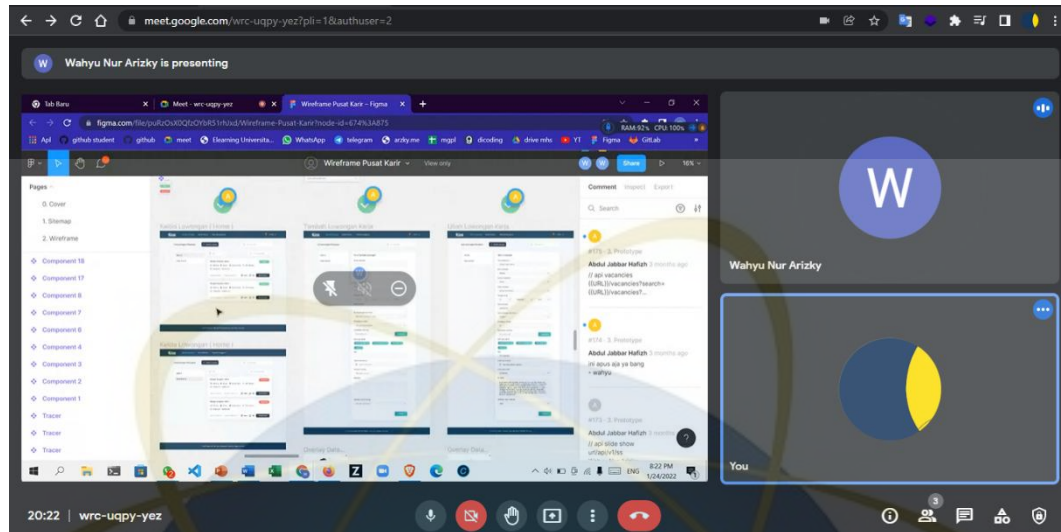
<https://ieeexplore.ieee.org/document/8016173/>

Utami, F. H., & Asnawati. (2015). *Rekayasa Perangkat Lunak* (J. Hutahaeen (Ed.); 1 ed.). DEEPUBLISH.

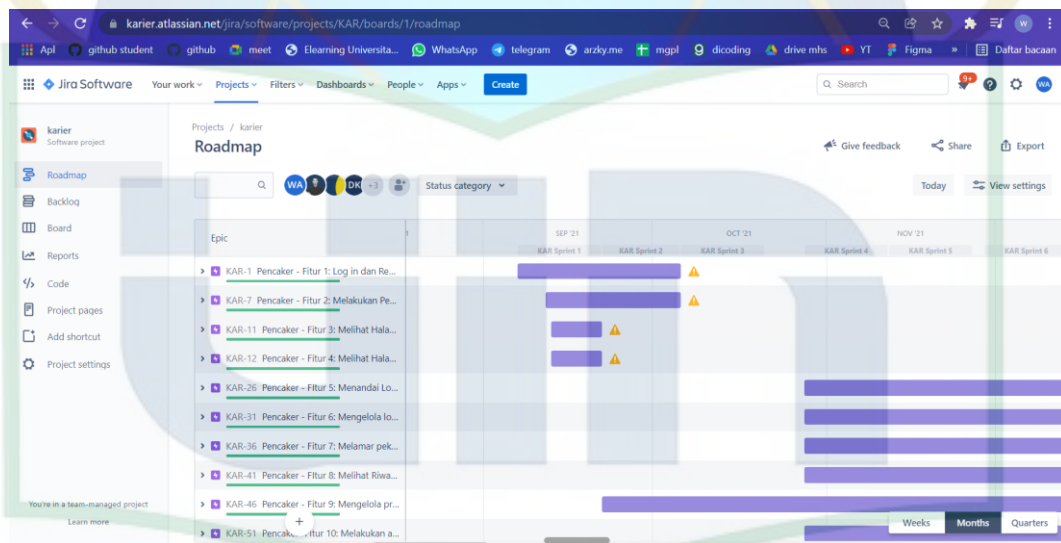
Vallon, R., da Silva Estácio, B. J., Prikladnicki, R., & Grechenig, T. (2018). Systematic literature review on agile practices in global software development. *Information and Software Technology*, 96(April 2017), 161–180.  
<https://doi.org/10.1016/j.infsof.2017.12.004>



## LAMPIRAN

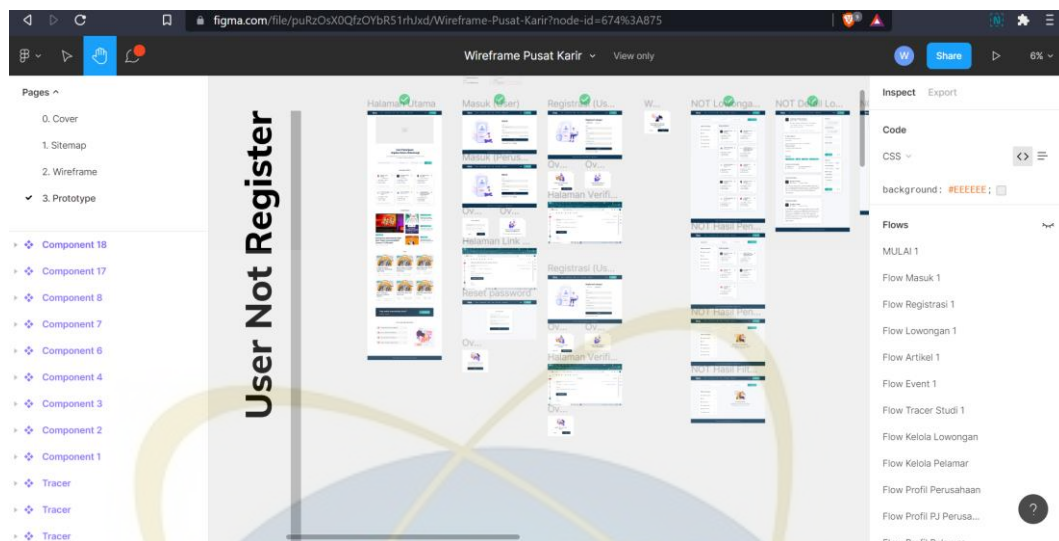


Gambar 16. Daily Standup

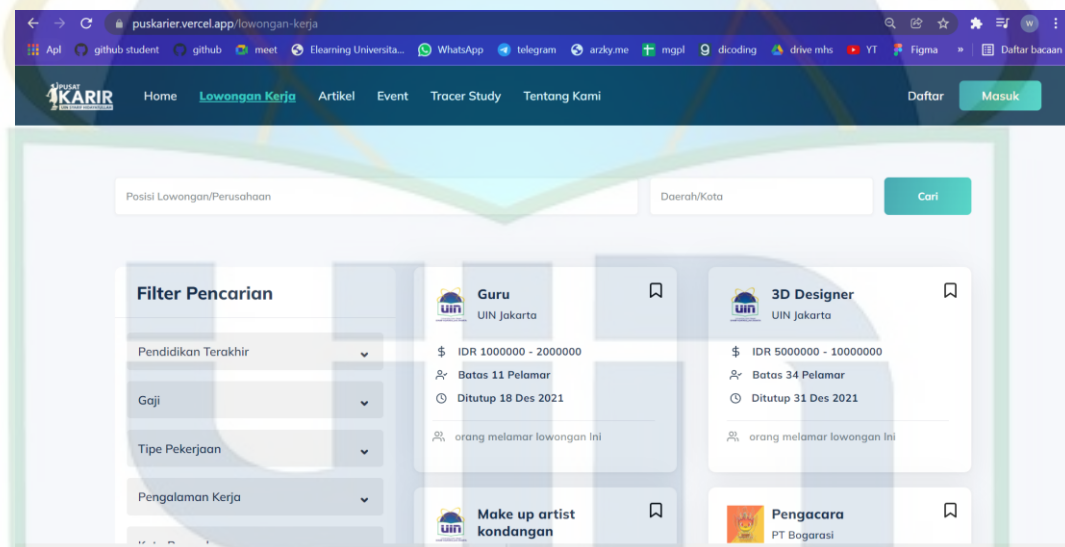


Gambar 17. Jira



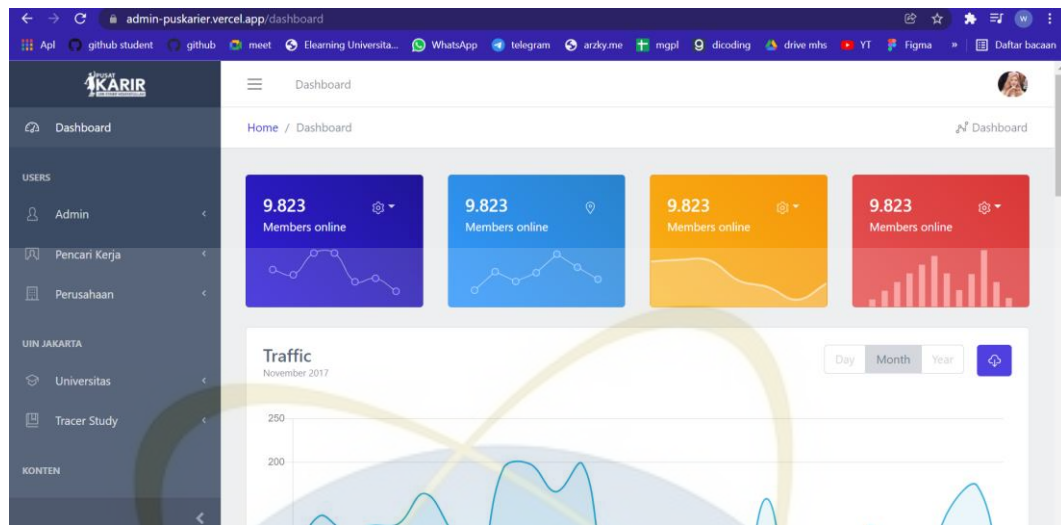


Gambar 18. Figma

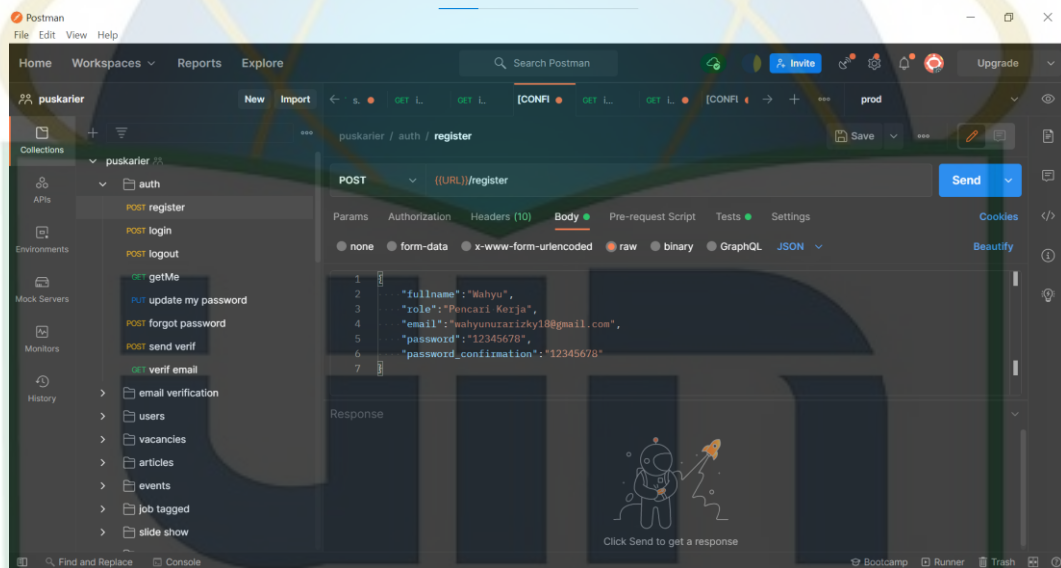


Gambar 19. Tampilan Pencaker &amp; Perusahaan

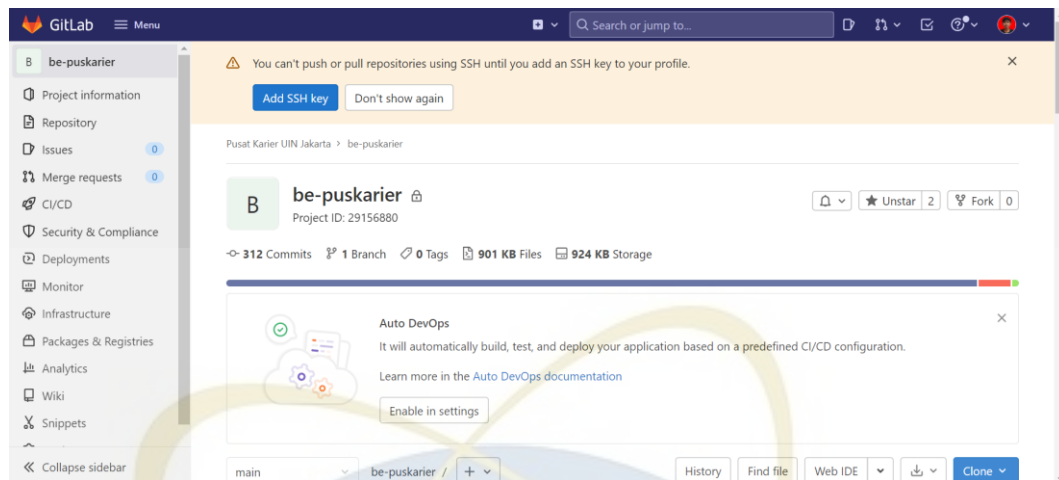




Gambar 20. Tampilan Admin



Gambar 21. Dokumentasi API menggunakan Postman



Gambar 22. Gitlab