

## BAB 3 METODOLOGI PENELITIAN

### 3.1 Alur Penelitian

Dalam penelitian ini, terdapat beberapa tahapan dalam metodologi penelitian, diantaranya adalah:

#### 1. Identifikasi Masalah

Langkah pertama yang dilakukan dalam melakukan penelitian adalah mengidentifikasi masalah yang ada. Dalam kasus ini, masalah yang ada merupakan seniman *graffiti* yang tidak bisa mengekspresikan dirinya karena keterbatasan regulasi.

#### 2. Studi Literatur

Berdasarkan masalah yang ada, dilakukan studi literatur berdasarkan penelitian-penelitian terdahulu untuk melakukan validasi atas landasan-landasan teori yang digunakan.

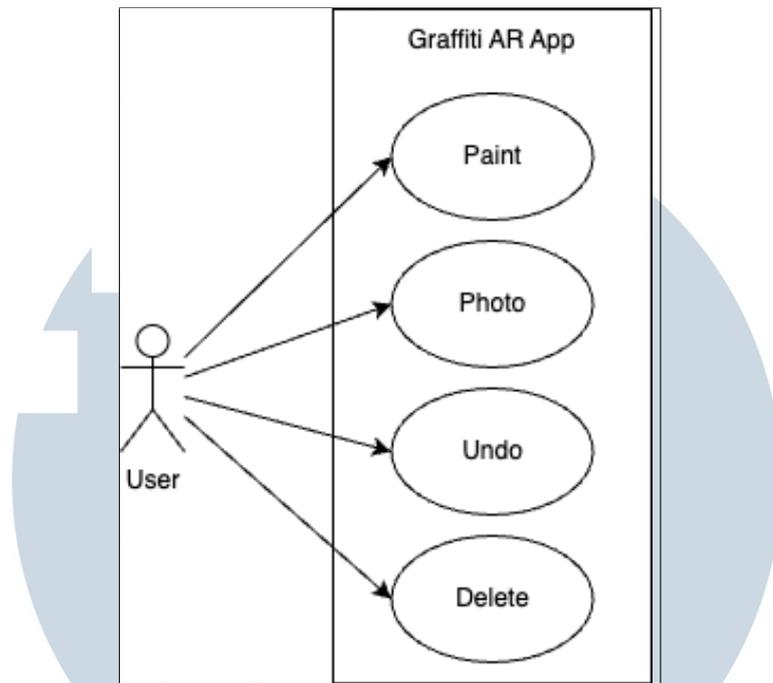
#### 3. Perancangan Aplikasi

Pada tahap perancangan dan pembangunan aplikasi, terdapat beberapa tahap yang dilakukan, mulai dari menentukan *flow* dari aplikasi dan juga membuat mock-up dari aplikasi untuk menentukan hirarki *UI* aplikasi. Aplikasi dirancang menggunakan metode *markerless AR* dan menggunakan *plane detection* sebagai perantara untuk mendeteksi tembok vertikal.

##### (a) UML

*Flow* dari aplikasi dibuat menggunakan *UML Use Case Diagram* dan *UML Activity Diagram*. Untuk gambaran *flow* utama dari aplikasi dengan *Use Case Diagram* dapat dilihat pada Gambar 3.1 yang menjelaskan tentang fitur-fitur yang dapat diakses oleh *user*.

##### i. Use Case Diagram



Gambar 3.1. *Use Case Diagram* dari aplikasi

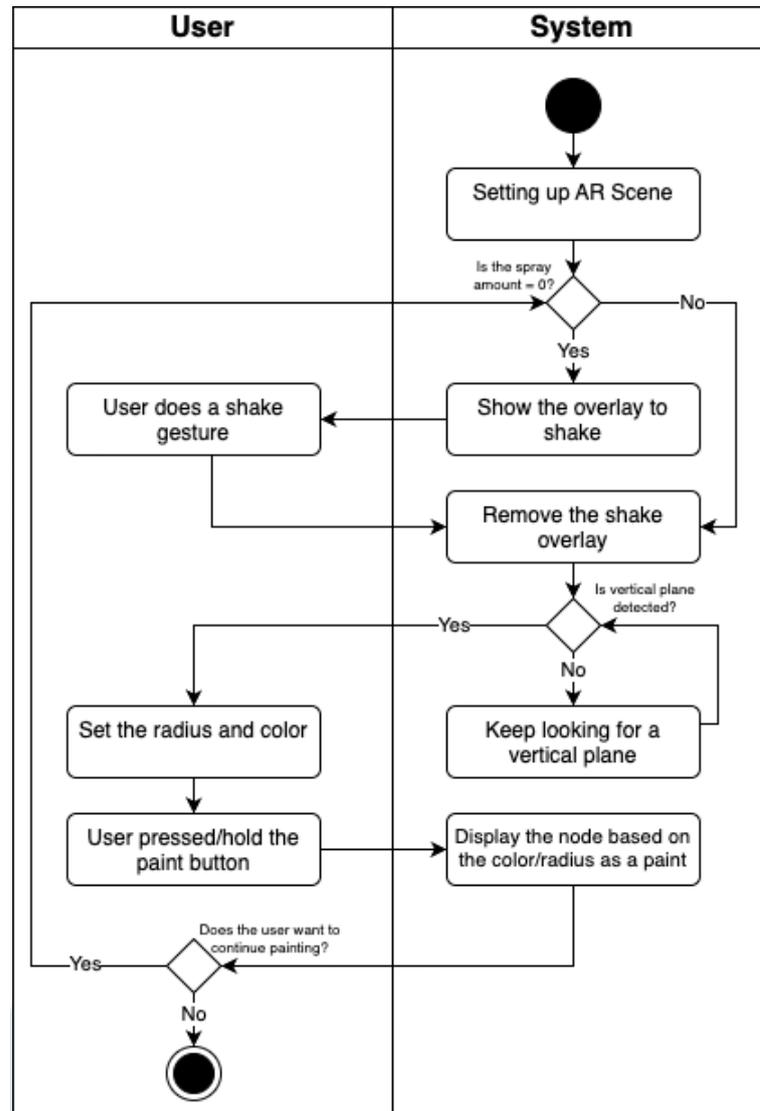
Seperti yang tertera pada Gambar 3.1, fitur-fitur yang dapat diakses oleh user pada aplikasi adalah fitur menggambar, fitur mengambil foto, fitur *undo*, dan fitur *delete*. Fitur-fitur tersebut akan dijelaskan pada sub-bab selanjutnya yang menjelaskan mengenai *activity diagram* dari tiap fitur.

ii. Activity Diagram

Untuk menjelaskan masing-masing fitur, digunakan *UML Activity Diagram*. Untuk penjelasan dari masing-masing fitur, dapat dilihat pada gambar-gambar yang akan ditampilkan dan dijelaskan pada bagian berikut:

A. Paint Activity Diagram

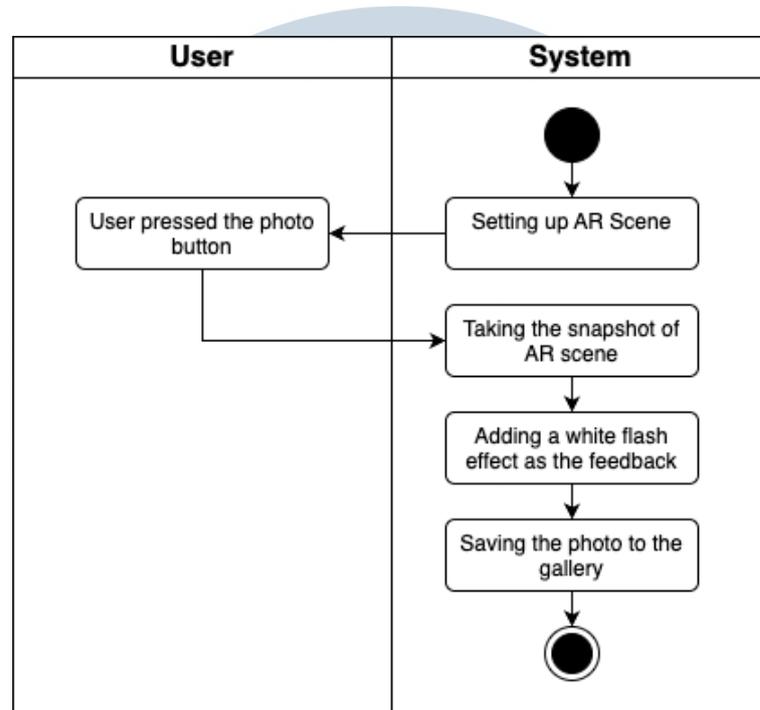
Untuk *flow* dari fitur *paint* akan dijelaskan melalui *activity diagram* pada Gambar 3.2.



Gambar 3.2. Activity Diagram untuk fitur *Paint* dari aplikasi

Pada Gambar 3.2, dijelaskan bahwa fitur *paint* pada aplikasi dilakukan seperti menggambar *graffiti* di dunia nyata, dimana *user* harus melakukan gestur mengguncang device untuk melakukan *reload* terhadap tinta untuk menggambar. Selain itu, user harus mencari bidang vertikal seperti tembok untuk menggambar *graffiti* yang diinginkan. Tidak bisa menggambar di sembarang tempat, dikarenakan apabila dilihat dari dunia nyata, *graffiti* kebanyakan digambar di tembok vertikal. Di bidang vertikal yang sudah terdeteksi, *graffiti* akan digambar menggunakan *node* yang disediakan oleh *ARKit*

## B. Photo Activity Diagram



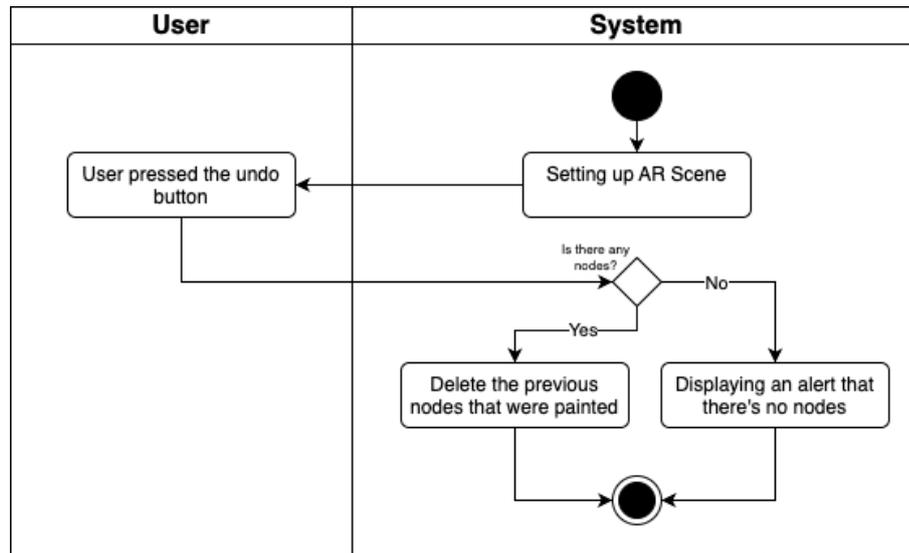
Gambar 3.3. Activity Diagram untuk fitur Photo dari aplikasi

Berdasarkan Gambar 3.3, untuk fitur foto, pertama-tama *scene* dari AR mengambil *snapshot*. Setelah itu, diberikan efek untuk memberi *feedback* ke *user* bahwa foto sudah diambil. Setelah itu, *snapshot* yang sudah diambil akan disimpan dan dapat dilihat di galeri.

## C. Undo Activity Diagram

Untuk *flow* dari fitur *undo*, akan dijelaskan pada *activity diagram* pada Gambar 3.4.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

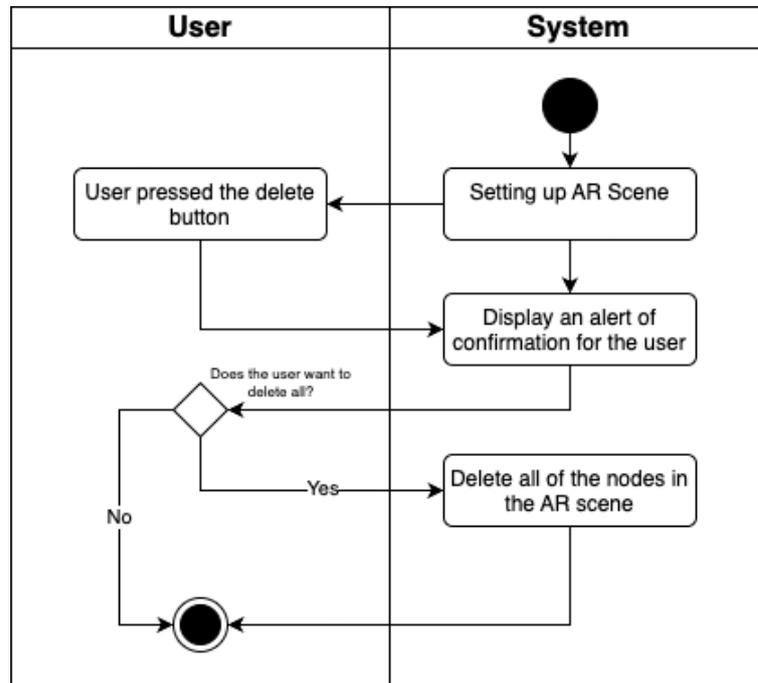


Gambar 3.4. *Activity Diagram* untuk fitur *Undo* dari aplikasi

Berdasarkan Gambar 3.4, untuk fitur *undo* hanya dapat dilakukan apabila *user* sudah menggambar sebelumnya. Apabila *user* belum menggambar, maka akan ditampilkan *alert* yang memberi tahu *user* bahwa *user* belum menggambar apapun pada *scene* AR.

#### D. Delete Activity Diagram

Untuk *flow* dari fitur *delete*, dapat dilihat pada *activity diagram* di Gambar 3.5.



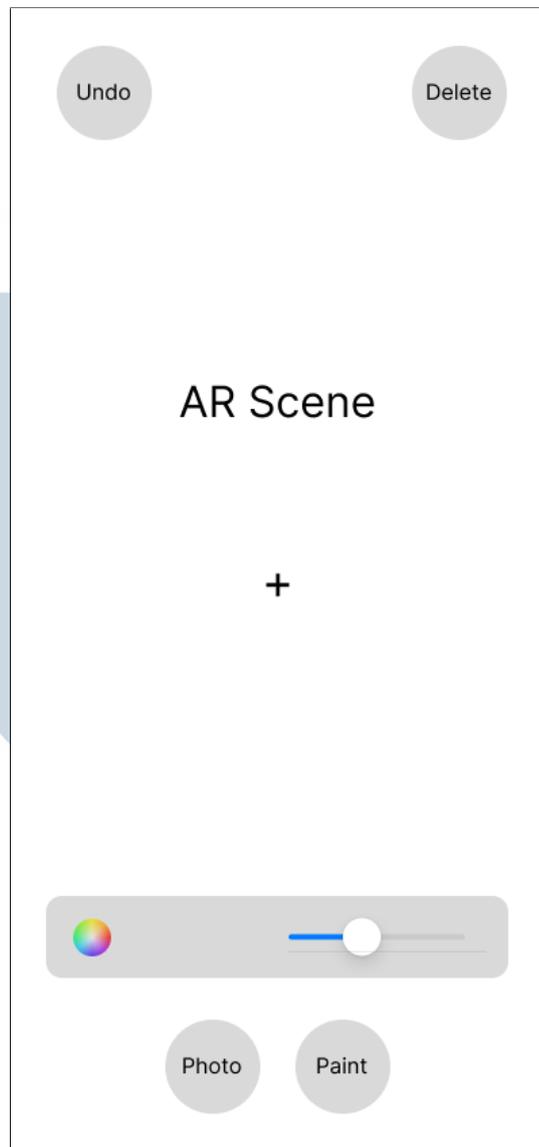
Gambar 3.5. Activity Diagram Delete dari aplikasi

Berdasarkan Gambar 3.5, fitur *delete* hampir sama dengan fitur *undo*. Namun, yang membedakan adalah di fitur ini, semua yang sudah digambar oleh *user* akan dihapus semua. Oleh karena itu, terdapat konfirmasi terlebih dahulu mengenai tindakan ini, karena bisa saja user tanpa sengaja menekan tombol *delete* dan menghapus semua gambar yang sudah dibuat.

(b) Mock-up

Dalam perancangan aplikasi, selain *flowchart* untuk menentukan flow dari aplikasi, dibuat juga *mock-up* dari aplikasi untuk melihat gambaran besar dari UI yang akan dibuat. Untuk *mock-up* dari aplikasi, dapat dilihat pada Gambar 3.6.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.6. *Mock-up* dari aplikasi

Berdasarkan Gambar 3.6, untuk tombol *undo* dan *delete* dapat diakses pada *toolbar* aplikasi yang akan digunakan untuk menghapus gambar yang sudah dibuat. Selain itu, untuk tombol foto dan tombol untuk menggambar ditaruh bersebelahan, serta fitur kustomisasi *node* (warna, radius) dapat diakses di atasnya.

Selain itu, terdapat tanda "+" ditengah-tengah layar. Tanda tersebut berfungsi untuk memudahkan pengguna mengetahui dimana mereka menggambar.

#### 4. Pembangunan Aplikasi

Aplikasi dibangun dengan menggunakan *xCode* sebagai *code editor* dan bahasa pemrograman *Swift* sebagai bahasa pemrograman yang digunakan.

#### 5. Pengujian Aplikasi

Aplikasi diuji dengan melakukan *deploy* aplikasi ke *TestFlight*, yaitu salah satu platform untuk melakukan *user testing* dan mendapatkan *feedback* apabila terdapat *feedback* dari pengguna, dan juga melalui *demo* yang diunggah ke platform *youtube*

#### 6. Pengumpulan Data

Metode pengumpulan data yang digunakan adalah metode kuantitatif, yaitu metode penelitian yang menggunakan teknik statistik, matematika, atau komputasi untuk mengumpulkan dan menganalisis data numerik [38]. Menurut Sugiyono (2013), metode kuantitatif adalah penelitian yang berlandaskan pada sifat positivisme untuk meneliti populasi atau sampel tertentu dan pengambilan sampel secara acak dengan pengumpulan data menggunakan skor atau angka, yang dapat di analisa dengan statistik [39].

Oleh karena itu, pengumpulan data dilakukan dengan melalui kuesioner yang dibuat menggunakan metode SUS, serta perhitungan persentase indeks kepuasan pengguna berdasarkan Skala *Likert*. Untuk jumlah responden, minimal jumlah responden menurut Sugiyono (2013) adalah 30 sampai 500 orang [39].

#### 7. Evaluasi

Melakukan *review* mengenai data yang sudah dikumpul dan melakukan perhitungan indeks kepuasan untuk dijelaskan di Bab 4.

#### 8. Dokumentasi

Menampilkan hasil-hasil responden, dan juga kode yang sudah diprogram untuk menjelaskan maksud dari kode-kode tersebut.