

## 1. Pendahuluan

### Latar Belakang

*Software Engineering* merupakan sebuah pembelajaran aplikasi dari sebuah pendekatan yang bersifat sistematis, disiplin, dan kuantifiabel dalam pengembangan maupun pembuatan software [1]. *Software Engineering* setiap tahun mengalami perkembangan dari semua sisi, salah satunya adalah semakin bertambahnya komponen didalamnya sehingga diperlukan sebuah teknologi agar komponen tersebut saling terkoneksi.

Banyak cara yang dapat digunakan agar suatu sistem dapat berkomunikasi dengan sistem lainnya. Salah satunya adalah dengan menggunakan teknologi *RestFul API (Application Programming Interface)*. *RestFul API* merupakan sebuah sistem *web service* yang memiliki kemampuan untuk meng-generalisasi layout yang didokumentasikan menggunakan metode API agar mudah dipelajari dan sebagai acuan terhadap fitur yang ada [2]. Dengan *RestFul API* sebuah sistem dapat dibangun dengan lebih cepat karena dapat dibagi kedalam beberapa sistem yang lebih kecil kemudian diintegrasikan sehingga sistem tersebut dapat saling berkomunikasi. Selain itu *RestFul API* juga dinilai *user friendly* dan mudah dipahami oleh *developer* untuk membuat *code* diatasnya [3].

Karena kemudahan dalam pembuatan dan penggunaan *RestFul API* saat ini banyak *developer* yang menggunakannya. Meskipun banyak digunakan, bukan berarti tidak ada masalah dalam implementasinya. Permasalahan yang sering muncul pada saat implementasi *RestFul API* adalah desain dan arsitektur API yang disediakan oleh penyedia API tidak sesuai dengan pengguna API. Sehingga diperlukan modifikasi pada user yang dimana akan memerlukan waktu[4].

Untuk mengatasi masalah tersebut dilakukan penelitian pembuatan API dengan menggunakan menggunakan metode *Human Centered Design (HCD)*. *Human Centered Design (HCD)* adalah proses desain yang berfokus pada kebutuhan pengguna baik itu sebagai individu maupun komunitas [5]. Alasan utama pemilihan *Human Centered Design* adalah dikarenakan *Human Centered Design* berfokus pada pendekatan pengguna. Dengan begitu sistem yang dibangun dapat sesuai dengan kebutuhan dan keinginan pengguna. Adapun metode yang digunakan adalah *personas*. Namun metode ini juga memiliki kekurangan. Salah satunya adalah bentuk desain API yang sangat bergantung pada kemampuan user, sehingga kebutuhan antar setiap pengguna memiliki perbedaan yang dapat dilihat dari keterampilan pengguna mengimplementasikan API. Namun kita juga dapat mengetahui kelebihan dan kekurangan dari masing masing metode.

Penelitian ini berkontribusi pada pengembangan *RestFul API*. Diharapkan dengan adanya penelitian ini dapat membantu pengembangan *RestFul API* yang sesuai dengan kebutuhan pengguna. Namun penelitian ini juga memiliki kekurangan. Yaitu metode yang digunakan hanya *personas*.

### Rumusan Masalah

Rumusan masalah dari penelitian ini adalah sebagai berikut :

- a. Apa kelebihan dan kekurangan penggunaan metode *persona* dalam implementasi *RestFul API Design*?
- b. Metode mana yang lebih baik dalam pembangunan *RestFul API* ?

### Tujuan

Tujuan penelitian ini adalah untuk mengetahui Keuntungan dan kerugian dari metode *persona* dalam pembuatan *API Usability* berdasarkan *Human Centered Design*. Sehingga diharapkan API yang disediakan oleh *developer* sesuai dengan kebutuhan dan keinginan pengguna.