BAB 1
PENDAHULUAN

1.1 Latar Belakang

Bandung merupakan kota metropolitan yang terletak di wilayah Jawa Barat. Kota ini memiliki julukan *Paris Van Java* dan Kota Kembang. Bandung bukan hanya kota yang menjadi rumah bagi jutaan warganya tapi juga menjadi kota yang kaya akan wisata bagi para pendatang. Sejak dahulu Bandung terkenal akan keindahannya. Kota yang dikelilingi oleh pegunungan ini memang dikenal memiliki berbagai hal menarik yang dapat menjadi daya tarik bagi wisatawan mulai dari tempat rekreasi hingga *event-event* bisa ditemukan di Kota Bandung.

Sejalan dengan peningkatan sektor pariwisata, maka upaya-upaya untuk mempromosikan wisata-wisata di Bandung pun semakin gencar dilakukan, salah satunya dengan lebih memperkenalkan *event* wisata yang ada di Kota Kembang ini. *Event* wisata sangat beragam dan memiliki keunikan masing-masing. Dengan memperkenalkan *event* wisata kepada khalayak luas, maka perekonomian di daerah pun dapat meningkat.

Oleh sebab itu untuk lebih memperkenalkan kebudayaan dan keindahan Kota Bandung dan mendorong minat masyarakat untuk mendatangi perayaan *event* wisata, diperlukan media yang dapat mengoptimalkan penyampaian informasi kepada masyarakat luas. Maka dari itu dibangunlah sebuah aplikasi yang dapat membantu masyarakat untuk menemukan informasi tersebut, yakni dengan menggunakan aplikasi BaViT (Bandung Visit Travel) Aplikasi Generator Konten Pariwisata Modul Manajemen dan *Event* Wisata.

Dengan adanya aplikasi ini diharapkan dapat membantu meningkatkan pariwisata di Bandung dan memberikan informasi kepada masyarakat mengenai *event-event* wisata, serta dapat memfasilitasi masyarakat dalam mempromosikan *event* yang ada.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang sudah dijelaskan sebelumnya, maka dirumuskan beberapa masalah sebagai berikut :

1. Bagaimana cara mengelola data *event-event* wisata berbasis komputer?
2. Bagaimana cara menginformasikan *event-event* wisata yang ada di Bandung ?
3. Bagaimana cara memfasilitasi penyampaian informasi dan pendapat dari masyarakat mengenai *event* wisata ?

1.3 Tujuan

Berdasarkan rumusan masalah yang sudah dibuat, maka dibuatlah tujuan dari proyek akhir ini yaitu :

1. Membangun aplikasi *event* wisata yang memudahkan pengelolaan konten *event* berbasis komputer.
2. Membangun aplikasi yang dapat menginformasikan *event-event* pariwisata yang ada di Bandung.
3. Membangun aplikasi yang dapat menjadi portal informasi dan penyampaian pendapat terkait suatu *event*.

1.4 Batasan Masalah

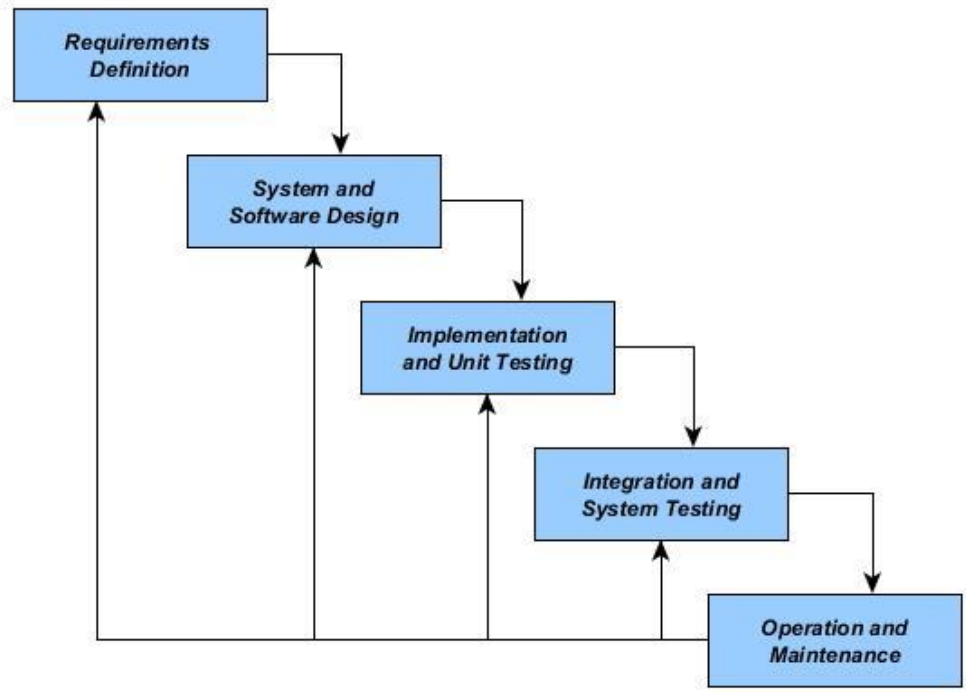
Batasan masalah dari proyek akhir ini adalah :

1. Modul ini hanya berbasis *website*,
2. Modul ini tidak menggunakan fitur *Google Map.*

1.5 Metode Pengerjaan

Metode pengerjaan digunakan dalam membangun aplikasi ini adalah model *System Development Life Cycle (SDLC)* tipe *Waterfall*, dikarenakan model ini merupakan model pembangunan perangkat lunak yang dibuat secara terstruktur, yang dinilai lebih sederhana dan cocok untuk aplikasi yang akan dibangun [1]. Adapun tahap-

tahap pengembangan perangkat lunak model *Waterfall* dapat dilihat pada gambar 1.1 sebagai berikut :



Gambar 1. 1 Metode *Waterfall*

Berdasarkan gambar 1.1, hal-hal yang harus dilakukan adalah :

1. *Requirements Definition*

   Requirements Definition adalah tahapan penetapan fitur, kendala dan tujuan sistem melalui konsultasi dengan pengguna sistem. Semua hal tersebut akan ditetapkan secara rinci dan berfungsi sebagai spesifikasi sistem [1]. Pada tahap ini dilakukan observasi melalui internet dan menyebarkan kuesioner kepada target pengguna yaitu masyarakat. Pemodelan bisnis yang berjalan menggunakan *Business Process Model and Notation (BPMN)*.

2. *System and Software Design*

   System and Software Design adalah proses pembuatan rancangan arsitektur sistem setelah proses pendefinisian kebutuhan dilakukan. Selain itu juga dilakukan identifikasi dan penggambaran terhadap abstraksi dasar sistem perangkat lunak beserta hubungan-hubungannya [1]. Pada tahap ini

menggambarkan sistem usulan aplikasi menggunakan *Business Process Model and Notation (BPMN)* menggunakan model sistem yang akan dibangun menggunakan *Unified Modelling Language (UML)* yang terdiri dari *Usecase diagram*, *Scenario Usecase*, *Class Diagram* dan *Sequence Diagram*, menggambarkan perancangan Basis Data dengan menggunakan *Entity Relationship Diagram (ERD)*, Skema Relasi serta *Class Diagram*.

3. *Implementation and Unit Testing*

   Implementation and Unit Testing adalah proses pembuatan kode dari desain yang akan direalisasikan sebagai satu set atau unit program [1]. Setiap unit akan diuji apakah sudah sesuai antara sistem yang dibangun dengan rancangan sistem yang dibuat sebelumnya. Dalam tahap ini pengkodean dilakukan dengan menggunakan Bahasa pemrograman *Hypertext Preprocessor (PHP)* dan *Javascript*, *Framework CodeIgniter (CI)* dan *MySQL* sebagai sistem manajemen basis data. Sedangkan pengujian dilakukan menggunakan metode *BlackBox Testing* dan *User Acceptance Test (UAT).*

4. *Integration and System Testing*

   Integration and System Testing merupakan tahap penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian ini dilakukan untuk mengetahui apakah *software* yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau tidak [2].

5. *Operation and Maintenance*

   Operation and Maintenance merupakan tahap terakhir dalam model *waterfall*, *software* yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru [2].

1.6 Jadwal Pengerjaan

Jadwal pengerjaan aplikasi selama analisis, perancangan, pengkodean aplikasi serta pengujian aplikasi dijelaskan pada tabel 1.1 sebagai berikut :

Tabel 1. 1 Jadwal Pengerjaan

No	Tahapan/Aktivitas	Maret				April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Analisis Aplikasi Sejenis																				
2	Membuat Fungsionalitas																				
3	Merancang Fungsionalitas																				
4	Membuat Diagram Use Case																				
5	Membuat Program																				
6	Membuat Class Diagram																				
7	Membuat Sequence Diagram																				
8	Melakukan testing																				
9	Dokumentasi																				