

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pemrograman merupakan kemampuan yang wajib dimiliki oleh setiap mahasiswa di jurusan teknik informatika, dimana pemrograman adalah proses pengimplementasian urutan langkah untuk menyelesaikan suatu masalah dengan menggunakan suatu bahasa pemrograman [1]. Bahasa pemrograman adalah prosedur atau tata cara penulisan program [1]. Untuk meningkatkan kemampuan pemrograman yang mahasiswa miliki yaitu dengan cara melakukan berbagai latihan dalam pengekseskuan kode program dari setiap bahasa pemrograman yang dimiliki. Setiap bahasa pemrograman memiliki struktur bahasa yang berbeda-beda oleh karena itu latihan pengekseskuan program secara berkala menjadi salah satu kunci utama dalam meningkatkan kemampuan atau *skill programming* yang dimiliki oleh seseorang khususnya mahasiswa.

Dalam dunia pendidikan pembelajaran pemrograman dapat dilakukan dengan metode *problem solving*, dari penelitian sebelumnya yang berjudul Penerapan Metode *Problem Solving* untuk Meningkatkan Aktivitas dan Hasil Belajar menyebutkan bahwa metode ini merangsang pengembangan kemampuan berpikir siswa secara kreatif dan menyeluruh, karena dalam proses belajarnya siswa banyak berlatih untuk menyoroti permasalahannya dari berbagai segi dalam rangka mencari pemecahan masalah tersebut, dimana dengan metode ini mahasiswa dituntut untuk menyelesaikan permasalahan dengan membuat sebuah

barisan kode program yang sesuai dengan masalah yang diberikan [2]. Kemudian untuk melakukan evaluasi terhadap kemampuan mahasiswa maka dilakukanlah sistem penilaian.

Sistem penilaian yang efektif dan efisien dalam pengecekan kode program yaitu menggunakan *autograder* hal ini dibuktikan dari penelitian sebelumnya yang menyebutkan bahwa *autograder* telah terbukti menjadi hakim yang efektif dan efisien dalam kontes pemrograman pada tingkat Internasional [3]. *Autograder* adalah suatu aplikasi atau modul program untuk melakukan penilaian atau *assessment* secara otomatis. Salah satu contoh sistem *autograder source code* untuk Java, C++ dan Python yang diintegrasikan dengan *CodeMirror* dapat diaplikasi dalam dunia pendidikan yaitu D&G [4]. Dari penelitian internasional yang membahas mengenai *autograder* belum ada yang mengembangkan *autograder* untuk bahasa pemrograman PHP [3] [4] [5] [6] [7] [8] [9] [10]. Sedangkan penelitian pada tingkat nasional yang berjudul Pengembangan Sistem Manajemen Pembelajaran Pemrograman Bahasa *Scheme, Java, PHP, dan Perl* membahas mengenai sistem *autograder* untuk PHP, akan tetapi sistem penilaian pada penelitian tersebut dianggap masih memiliki beberapa kekurangan diantaranya sistem penilaian tidak memiliki sistem koreksi, dan tidak mengkategorikan penilaian terhadap modul pemrograman yang dibahas seperti penggunaan operator aritmatika, pengulangan, pemilihan dan lain-lain [11]. Oleh sebab itu, maka dibuatlah sebuah penelitian yang berjudul “Modul *Autograder* untuk Bahasa Pemrograman PHP” dengan menggunakan sistem penilaian *test case* yang dilengkapi dengan sistem *real time feedback* atau sistem koreksi terhadap kesalahan pengetikan *grammar* PHP yang dilakukan oleh *user*, yang dapat

memberikan informasi mengenai letak kesalahan yang dilalukan oleh *user* dalam penulisan kode program. Selain itu sistem penilaian dalam autograder ini dilengkapi dengan mengkolaborasikan nilai kategori yang sesuai dengan modul pemrograman yang dibahas.

## 1.2 Rumusan Masalah

Penelitian ini mengkaji mengenai proses pengecekan dan penilaian sebuah *source code* PHP secara otomatis. Masalah - masalah yang akan dikaji dalam penelitian ini adalah :

1. Bagaimana menentukan sebuah skema penilaian *source code* PHP ?
2. Bagaimana memberikan *feedback source code* secara *real time* ?
3. Bagaimana membuat sebuah sistem penilaian *source code* PHP secara otomatis ?

## 1.3 Batasan Penelitian

Pada penelitian ini ditetapkan beberapa batasan masalah diantaranya :

1. *Source code* yang akan dinilai dalam sistem ini hanya *source code* yang menggunakan bahasa Pemrograman PHP.
2. Sistem penilaian yang digunakan dalam *autograder* ini menggunakan sistem *testcase* yang dikolaborasikan dengan nilai penggunaan token PHP berdasarkan kategori soal.
3. Sistem ini menghasilkan *output* penilaian *source code* PHP yang dikelompokkan menjadi berbagai kategori yaitu sukses (*match*) dan *didn't match*.

#### 1.4 Tujuan Penelitian

Tujuan yang hendak dicapai dalam penelitian ini adalah :

1. Menentukan skema penilaian terhadap *source code* PHP
2. Memberikan koreksi terhadap kesalahan *source code* yang dilakukan secara *real time*.
3. Membuat sebuah sistem penilaian *source code* PHP secara otomatis.

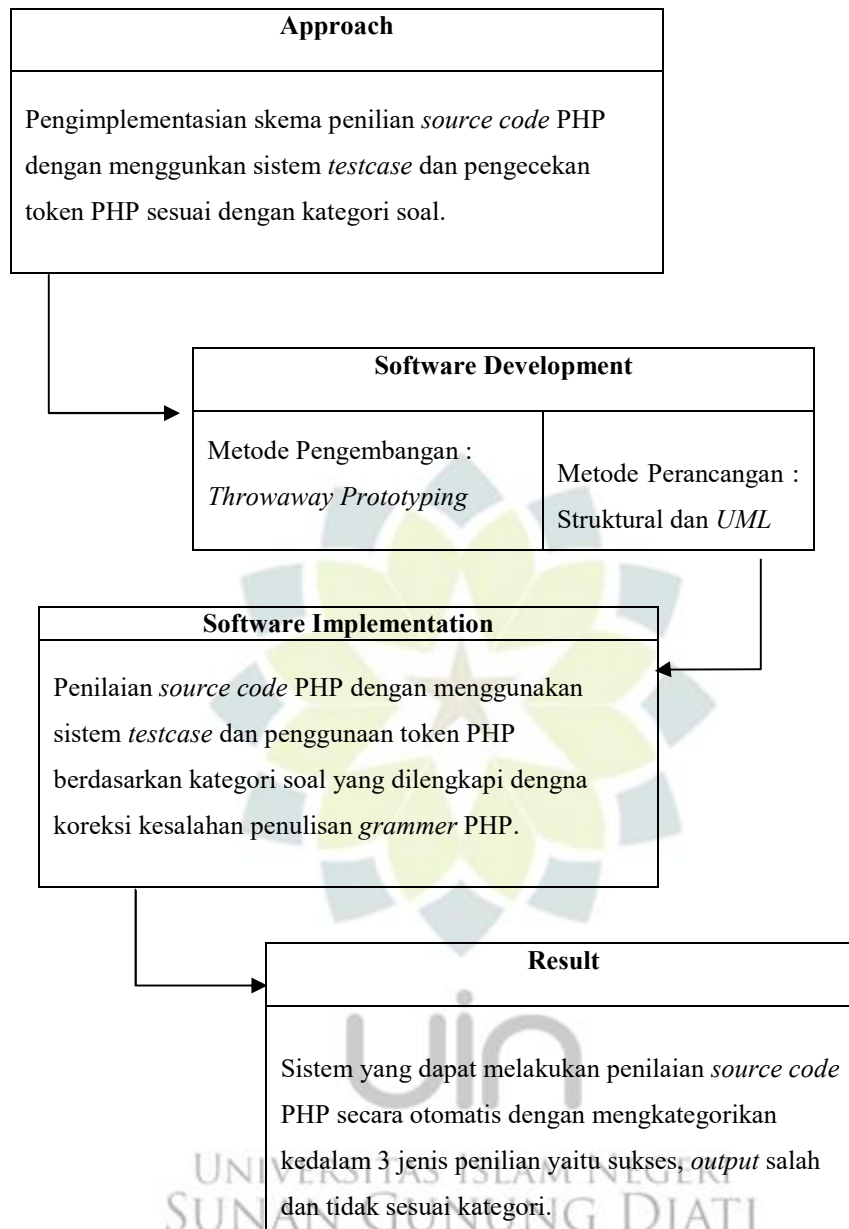
#### 1.5 Manfaat Penelitian

Dengan adanya penelitian ini maka penilaian terhadap *source code* dan koreksi program dengan kode program yang menggunakan bahasa pemrograman PHP dapat lebih efektif dan efisien sehingga tingkat keterampilan pemrogramannya dapat terlatih untuk menyelesaikan sebuah masalah dalam pemrograman PHP.

#### 1.6 Kerangka Pemikiran

Berdasarkan tinjauan pustaka yang telah dijelaskan maka kerangka pemikiran dapat diuraikan sebagai berikut yang terdapat dalam gambar 1.1 di bawah ini:





Gambar 1.1 Kerangka Pemikiran Penelitian

## 1.7 Metodologi Penelitian

Metodologi yang digunakan dalam penelitian kali ini adalah sebagai berikut :

### 1.7.1 Teknik Pengumpulan Data

Metode yang digunakan pada pengumpulan data yang dilakukan kali ini terdiri dari 2 tahapan, yaitu :

1. Observasi

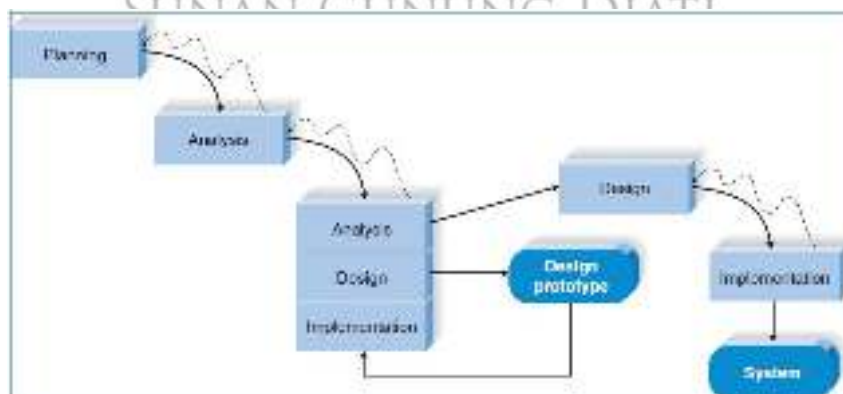
Teknik pengumpulan data dengan mengadakan penelitian dan peninjauan langsung terhadap permasalahan yang diambil.

2. Studi Literatur

Pengumpulan data dengan cara mengumpulkan literatur, jurnal, *paper* dan bacaan-bacaan yang ada kaitannya dengan judul tugas akhir.

### 1.7.2 Metodologi Pengembangan

Metodologi pengembangan perangkat lunak yang digunakan yaitu *throwaway prototyping model*. Proses pembangunan sistem menggunakan model *throwaway prototyping* ini terdiri dari *planing*, *analysis* (*analysis*, *design*, *implementation*), *design* dan *implementation* [12].



Gambar 1. 2 *Throwaway Prototyping model* [12]

Dari skema model *throwaway prototyping* pada gambar 1.2 di atas menggambarkan bahwa model ini merupakan pengembangan dari metodologi *system prototype* dimana dalam model ini memiliki tahapan analisis yang cukup teliti. Setiap sistem ketubuhan yang telah direncanakan pada tahap perencanaan kemudian masuk kedalam analysis, pada tahap ini setiap kebutuhan sistem di analisis untuk kebutuhan pengembangan fitur-fitur dalam sistem yang diimplementasikan kedalam *design prototype*. Proses ini memberikan kemudahan terhadap pengembang dalam menyampaikan informasi terhadap *client* sebelum melakukan tahap *development* yang akan diimplementasikan kedalam *real system* [12].

### **1.8 Sistematika Penulisan**

Untuk memudahkan dalam melakukan penyusunan skripsi ini maka dibuatlah sebuah sistematika penulisan yang terdiri dari bagian bagian berikut ini :

#### **BAB I PENDAHULUAN**

Bab ini menguraikan latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, *state of the art*, kerangka pemikiran, metodologi penelitian dan sistematika penulisan.

#### **BAB II LANDASAN TEORI**

Bab ini membahas mengenai teori-teori yang digunakan untuk memperkuat pernyataan- pernyataan yang berkaitan dengan judul skripsi yang berkolerasi dengan teori yang telah ada. Selain itu dalam bab ini menjelaskan berbagai teori – teori pendukung untuk digunakan penyusunan skripsi ini.

### **BAB III ANALISIS DAN PERANCANGAN**

Bab ini menjelaskan mengenai analisis dan perancangan sistem yang akan dibuat yang diawali dengan penjelesan mengenai sistem yang berjalan dan kemudian melakukan analisis terhadap sistem tersebut. Selanjutnya membuat tentang desain sistem, basis data maupun mengenai *interfacenya*. Pada bab ini analisis dan perancangan yang diuraikan adalah analisis sistem operasi, arsitektur sistem, perancangan sistem, perancangan basis data, dan perancangan aplikasi.

### **BAB IV IMPLEMENTASI DAN PENGUJIAN**

Bab ini akan menguraikan mengenai implementasi dari perancangan yang telah dibuat serta pengujian aplikasi yang telah dianalisa dan dirancang sebelumnya. Bab ini membahas mengenai perangkat lunak pembangun, perangkat keras pembangun, implementasi basis data, implementasi antar muka, pengujian perangkat lunak.

### **BAB V PENUTUP**

Bab ini menjelaskan mengenai kesimpulan, usulan, solusi dan saran terhadap aplikasi yang telah dibangun.