

Laporan Tugas Besar 1

IF2121 Logika Informatika

Jonathan Christopher
(13515001, K-01)

Jordhy Fernando
(13515004, K-01)

Jauhar Arifin
(13515049, K-01)

Turfa Auliarachman
(13515113, K-01)

November 29, 2016

1 Deskripsi Program

Program yang kelompok kami buat untuk Tugas Besar I mata kuliah Logika Informatika ini bernama *Submerged*. *Submerged* adalah sebuah game petualangan berbasis teks yang diimplementasikan menggunakan bahasa pemrograman deklaratif Prolog. Dalam game ini, pemain harus mengeksplorasi sebuah kapal selam yang sedang tenggelam dan mencari jalan keluar ke permukaan. Pemain dapat melakukan perintah-perintah untuk berpindah tempat, mengambil dan menggunakan objek, serta berinteraksi dengan NPC.

2 Keunikan dan Kelebihan

Game *Submerged* ini memiliki beberapa keunikan:

- Game dikompilasi dari program Prolog menjadi sebuah file *executable* yang sudah memuat *interpreter* Prolog, sehingga game dapat langsung dijalankan tanpa terlebih dahulu menjalankan *interpreter* Prolog secara manual.
-

3 Peta Permainan dan Daftar Objek

Terdapat objek-objek statis maupun interaktif yang terletak pada beberapa lokasi dalam game ini:

kegunaan, letak

4 Penjelasan *Command*

Terdapat beberapa *command* yang dapat digunakan oleh pemain dalam game ini:

kegunaan, skenario penggunaan

5 Hasil Eksekusi

screenshot

6 Pembagian Kerja

Berikut adalah tabel pembagian tugas dalam kelompok:

Komponen	Anggota	Dikerjakan pada
E	expression	keseluruhan ekspresi aritmatika
T	term	ekspresi angka, hasil perkalian atau hasil pembagian

7 Source Code

7.1 main.pl

```
/* Main menu loop */

:- initialization(submerged).
submerged :- menuLoop.

menuLoop :-
    nl,
    write('=_SUBMERGED_='), nl,
    write('====='), nl,
    nl,
    repeat,
    catch((
        write('Enter_[start.]_to_begin,_[load.]_to_load_game,_[exit.]_to_quit:'), nl, write('>_')
    ),
        read(user_input, Input),
        menuAction(Input)
    ), error(syntax_error(_), _), (
        write('Invalid_input.'), nl, fail, !
    )).

menuAction('start') :-
    init_gameState, /* set initial game state */
    gameLoop.

menuAction(load(FileName)) :-
    loadGame(FileName),
    gameLoop.

menuAction('exit') :- abort.
menuAction('quit') :- abort.
menuAction(_) :- write('Invalid_action.'), nl, !, fail.

/* Main game loop */

gameLoop :-
    repeat,
    catch((

        /* Render game state */
        render_gameState,

        /* Input */
        write('>_'),
        read(user_input, Input),

        /* Process input */
        process(Input),

        \+ win,
        \+ gameOver,

        fail

    ), error(syntax_error(_), _), (
        write('Invalid_input.'), nl, fail, !
    )).
```