# To Err Is Human: Building an Automatic Error Generator System

Luiz Carlos Begosso(✉) and Marcos Roberto Alves Medeiros

Fundacao Educacional do Municipio de Assis – FEMA Assis, Assis, SP, Brazil
begosso@gmail.com, zxmarcos@gmail.com

**Abstract.** This paper introduces Human Error Generator (HEG), an erroneous human performance simulator which considers that the errors emerge from human-computer interaction. HEG, in its current version, has two implemented modules: one that designs the interface in which the simulated user performs a task; and an automatic human error generator module. The system is linked to the cognitive architecture ACT-R and is able to simulate the following errors during the user performance: perceptual confusion, omission, inversion, repetition and intentionality reduction. Furthermore, HEG generates a report with the error-affected human performance showing the information related to the cognitive process involved in the simulated task as output. This paper aims to contribute to the field of human-computer interaction regarding the development of systems that anticipate human error.

**Keywords:** Error generator · Human error · Human-computer interaction

## 1 Introduction

Human actions are an essential part of any task execution, particularly when dealing with computational systems. In various situations, even when the human operator is focused on adequately performing the designed task, systematic errors can be committed.

Therefore, efforts are being made to develop tools which can tolerate the occurrence of errors that may emerge from human-computer interaction (HCI). Programmers develop mechanisms for error protection and error recovery from human tasks, whether they are motor, perceptual or decision-making. Naturally, these same mechanisms of protection can increase the complexity of human-computer interaction.

Studies on human factors cannot consider human beings alone, as well as human error cannot be solely seen from the performance of humans, [1]. We believe that human factors investigations should take not only the people involved but also their work environment as the resources for task execution.

This work's goal is to present the current stage of development of Human Error Generator (HEG) software, which can contribute to the study of important consequences for operational reliability, helping human-computer interfaces programmers minimize the chances of errors occurring during the interaction.


C. Stephanidis (Ed.): HCII 2016 Posters, Part I, CCIS 617, pp. 93–96, 2016.
DOI: 10.1007/978-3-319-40548-3_15

## 2 The Human Error Generator System

In order to develop a flexible platform - graphical user interface (GUI) - for the simulation of human performance that takes into consideration the error, HEG looks like an interfaces editor, as shown in Fig. 1.

HEG is divided into two modules: the task design and registration module; and the automatic human error generator module.

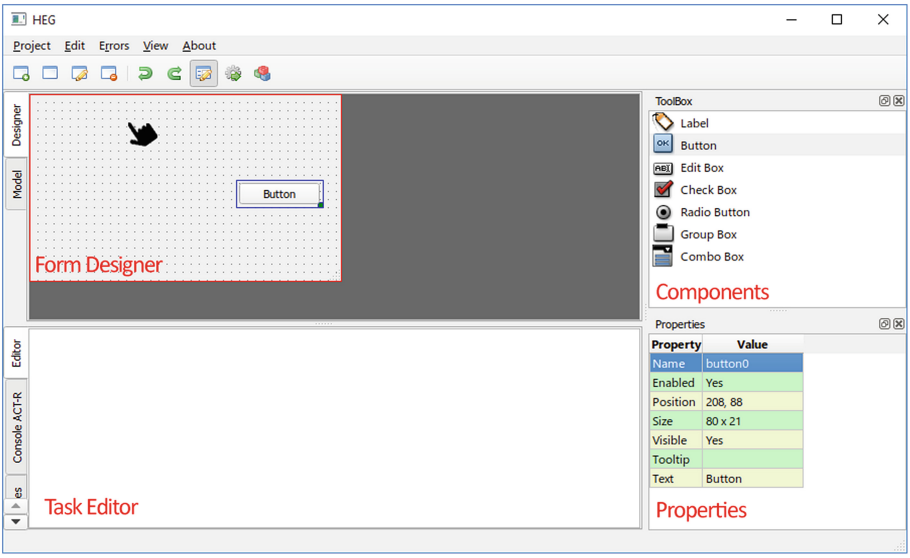


**Fig. 1.** Task description module

### 2.1 Task Design Module

In Fig. 1, at Components Box, the items that can be used by the operator to design the GUI are shown. The most common components in computer graphic interfaces were implemented in HEG, enabling the creation of diversified interfaces. Its operation method is based on "drag and drop".

The form is the representation of the GUI where components are arranged and can be subject to the following user operations: change, resize, move and delete. When a component is selected on the form, their properties automatically appear in the Property Editor box. The list of items in a check box can be changed within the form itself by double-clicking the component.

To describe a task it is necessary to use a domain specific language in which the steps to complete the task are informed by elementary actions on the GUI.

A specific language was elaborated for HEG: the Task Description Language (TDL) which to date consists of six reserved words: select, mark, clear, pick, press and

fill. In this context, the primary actions planned for HEG are: select radio button, check /uncheck the check box, press button, choose check box and fill the text box.

### 2.2 Automatic Error Generator Module

The design of HEG is supported by a human information processing model which aims to explain human behavior and is commonly used to study human error activities that were caused during problem-solving processes, [2]. The model is divided into three activity levels performed by a human operator at a certain task: the skill-based level, the rule-based level and the knowledge-based level.

The human error taxonomy adopted by HEG is the theoretical model denominated GEMS (Generic Error Modeling System) which aims to explain human errors, [3]. Such taxonomy is structured according to the performance levels proposed by [2].

HEG is currently prepared to generate skill-based level errors: perceptual confusion, omission, inversion, repetition and intentionality reduction. Also, HEG uses the cognitive architecture ACT-R for the simulation of human performance error. HEG's architecture is shown in Fig. 2.

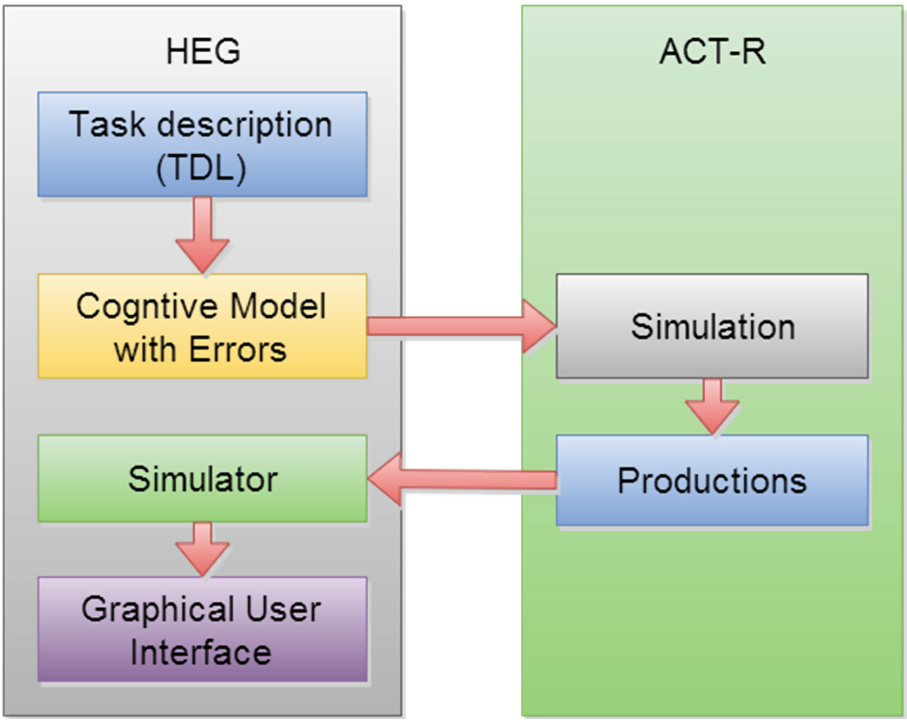


**Fig. 2.** HEG architecture

The outputs by HEG are models to the aforementioned architecture, i.e., the tasks described in TDL are translated to ACT-R models which can be altered with the errors requested by the operator in one of the steps in the task.

When executing the generated models, a trace is created with the information constituting the entire cognitive process involved in the task, according to ACT-R theory for real-time simulation of these errors.

## 3 Conclusion

Human errors may be considered contributing factors to a series of accidents and incidents in several areas where people interact with computer systems. Thus, this paper aimed to present HEG, a human performance simulator that takes into account errors that can occur during human-computer interaction.

A point to be made about HEG is its feature of generality, that is, the system can support the design of any user interface, regardless of the domain. On the other hand, the work here presented is prepared only for the automatic generation of skill-based errors. It is known, however, that the complexity of the human cognitive system can trigger user behavior that goes beyond the skill level. Therefore, as future work, we intend to expand the simulation of human errors, implementing rule-based level and knowledge-based level errors.

HEG design is based upon error generation, regardless of its the probability of occurrence. It is up to the programmer of HCI, while aware of the importance of these probability values, to analyze and reflect on the output of the system proposed here and the impact of the errors generated in the human-computer interaction.

**Acknowledgments.** The authors would like to acknowledge the support of Fundacao Educacional do Municipio de Assis (FEMA) and CNPq.

## References

1. Dekker, S.: The Field Guide to Understanding Human Error. Ashgate, Burlington (2006)
2. Rasmussen, J.: Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. IEEE Trans. Syst. Man Cybern. **13**(3), 257–266 (1983)
3. Reason, J.: Human Error. Cambridge University Press, Cambridge (1990)