



Application of a Continuous Improvement Method for Graphic User Interfaces Through Intelligent Systems for a Language Services Company

Osvaldo Germán Fernández^{1,2(✉)}, Pablo Pytel^{1,2(✉)},
and Ma Florencia Pollo-Cattaneo^{1,2(✉)}

¹ Programa de Maestría en Ingeniería en Sistemas de Información, Facultad Regional Buenos Aires, Universidad Tecnológica Nacional, Buenos Aires, Argentina

`ppytel@frba.utn.edu.ar`

² Grupo de Estudio en Metodologías de Ingeniería en Software (GEMIS), Facultad Regional Buenos Aires, Universidad Tecnológica Nacional, Buenos Aires, Argentina

Abstract. The Graphic User Interface represents the most common mechanism in human-computer interaction. Thus, its correct development is an utterly important task, which is reflected to the point where most software projects fail due to low user acceptance. The techniques, methodologies, tools, and principles to improve Graphical User Interfaces have been the object of interest of researchers and organizations over time. Due to the low success rate, a recent trend consists in experimenting with a greater number of interface alternatives, lowering their development costs through automation and Artificial Intelligence. Therefore, this paper presents a method that, by Interactive Genetic Algorithms, a semi-automatic continuous improvement process for landing pages is established. The proposed method is applied in a study case, consisting in a landing page that belongs to a translation services company. The method application provides positive results, enabling the landing page to reach the goals for which it was designed.

Keywords: Graphic User Interface · Landing page · Continuous improvement · Intelligent systems · Interactive Genetic Algorithms

1 Introduction

Even before the invention of writing, humanity has created interfaces that have served as support resources to satisfy their interaction needs [1], and over time, those have evolved alongside the cognitive human capacity [2]. With the advent of digital screens, a new notion of user interface was born, defined as the part of a computer that the user can see, hear or touch, or, in simpler terms, the one that the user can comprehend and control [3, 4]. This interface makes the user see what the product is capable of, which is why it is considered an essential part of any software application.

On the other hand, the Graphical User Interface (GUI) represents the most common mechanism in human-computer interaction, and given its prominence, it can influence

the software application success. This is why the correct construction of GUIs is a vitally important task, crossed by a wide diversity of disciplines and activities [5–7]. Ramírez et al. [5] emphasizes the link between GUIs and the software projects success, mentioning that about 70% fail due to low acceptance by users. Thus, it is correct to mention that a properly designed interface is vital for the success of the application that contains it [6]. Several studies support the relationship between the GUI quality and the credibility of the application [7], the perception of integrity [8], customer loyalty [9], the intention to revisit [10], and lower costs in customer acquisition and increases in retention [11].

In addition, the GUI acquires greater relevance due to its impact on usability and ergonomics, the latter being understood as usability oriented to physiological aspects in such as touch and mobile devices [12, 38]. For this reason, it has been described that about 67% of users are more likely to use interfaces adapted for mobile devices [13].

Lastly, the benefits of an appropriate GUI are recognized by several organizations that provide software worldwide, basing their decisions in interface design disciplines [14].

Due to these aspects, GUIs have been the main interest object of researchers and organizations over time. However, the improvement proposals that emerged from this interest have not ensured their success, which is reflected in the fact that 80% of the time designers are wrong about what the user really wants [15] and that, for example, only a third of the ideas implemented by companies like Microsoft have improved the metrics for which they were implemented [16]. To overcome these obstacles, experimenting with a great amount of GUI variations has been lately considered, which results in higher construction costs.

In this context, the present work proposes a method that, if followed by an IT Professional, facilitates the implementation of a continuous improvement process for GUIs through Artificial Intelligence and automation, and thus lowering their construction costs. For that purpose, Sect. 2 describes the proposed method and Sect. 3 presents the results of the implementation in a study case. These results are then analyzed in Sect. 4. Finally, Sect. 5 presents the conclusions and future lines of work.

2 Proposed Method

Based on the problems identified in the previous section, the proposed solution consists on a method that seeks to guide IT Professionals and UX designers in the implementation of a continuous improvement process for web platforms through Artificial Intelligence based on Interactive Genetic Algorithms (described in Sect. 2.1).

The detailed method and analysis, in this case, focus on landing pages. A landing page is a visual platform where organizations can attract visitors in a more comfortable way to make sales [17]. These pages are very suitable for continuous improvement processes, mainly because, in many cases, they represent the first audience interaction with the application. For example, the landing page presented for the study case (Sect. 3) belongs to a translation services company.

In order to evaluate the result of the interaction, it is necessary to use well-established metrics within the industry [18, 35, 36]. Therefore, it is then appropriate to analyze the

perception phenomenon in such platforms. The concept has been developed in various studies, analyzing aspects such as complexity [19, 20], symmetry [21], color [22], or several of these characteristics combined [6]. Likewise, user inherent characteristics such as sociodemographic variables have been described [23, 24]. However, for the present method, the approach proposed by Kohavi & Longbotham [15] is used, since this approach on perception can be measured through clearly defined indicators, which reflect the behavior of users on the web platform: conversion rate, sessions per user, session duration, bounce rate and pages visited per session.

The general structure of the proposed method is later described in Sect. 2.2.

2.1 Interactive Genetic Algorithms

A Genetic Algorithm builds on a population of individuals that represent possible solutions to a problem. Each of these individuals is maintained in the form of a “chromosome”, which is merely a string of characters that encodes a solution to a problem [27]. Mirjalili [28] describes the following stages for a Genetic Algorithm:

1. *Initial Population*: The genetic algorithm begins with a randomly generated population. This population includes multiple solutions.
2. *Selection*: Natural selection is the main inspiration for this algorithm, which is why a fitness function is used to assign a score to each individual, based on its performance in the environment. The best individuals are selected for the next generation.
3. *Crossover*: After selecting the individuals in the previous operation, two solutions (parents) combine their characteristics in two new solutions (children).
4. *Mutation*: It is the last evolutionary operator, in which one or multiple genes are altered in order to maintain the population diversity by introducing randomness.

On this basis, an Interactive Genetic Algorithm (also called IGA) includes human evaluation in the optimization process [29]. During the selection operation, a fitness function is used, in which the users interact with the generated solutions to assign a value. Although there are some examples of successful IGA implementations in the GUI improvement domain [30], most of them consist on proprietary tools with limitations considered by their authors. Many of these limitations are not from the tool itself, but on how the tool is defined and applied to meet the stated goal. For this reason, embedding the tool in an engineering process is imperative. In fact, the effective integration of activities and media (such as, in this case, an AGI software tool) is inherent to Information Systems Engineering [31]. To achieve this, it is possible to apply knowledge from different Engineering disciplines such as Information Systems Engineering [31], Software Engineering [32] and Requirements Engineering [33].

For this proposed method, one landing page variation is one possible solution, which is encoded in a chromosome whose structure is composed of three elements:

- a Permutation gene (GP), that defines the relative order of a web element, and therefore its position,
- a Color gene (GC), where three genes for each color, corresponding to tone -H-, saturation -S- and color luminosity -L-, and

- a Style gene (GE), that reflects various modifications, such as typographic hierarchies.

In addition, other elements are stored, such as the chromosome generation and the chromosome performance result.

2.2 General Structure of the Proposed Method

The proposed method describes phases and activities necessary to identify the landing page goals, determining GUI elements whose attributes are modified. Subsequently, the method lists the phases and activities in order to build tools that automate elements intervention, as well as build tools that automate the evaluation and deploy of these intervened GUIs, under two architectural proposals, which depends on the project needs.

All these activities are performed by two main roles:

- *IT Professional* (also referred to as IT): This role is the main responsible for the construction of all the software tools mentioned previously (and even responsible for tool acquisition if those exist and such is considered pertinent). It is for this reason that this role skills and experience are strongly oriented to web development. In addition, it is convenient that the IT Professional has experience in software methodologies application as well as requirements gathering and interpretation.
- *UX designer* (also referred to as UX, “User Experience designer” or “Interaction Expert”): it is understood as someone whose work domain is the ways in which users interact with and through computers [25]. This broad definition considers that the role includes graphic visual interaction disciplines (such as graphic design) but exceeds it, including also holistic ergonomics knowledge and even product and business ideation [26]. In fact, under the presented method, the UX designer is responsible for identifying organizational goals and understanding in which ways the landing page collaborates with them. This information is essential to determine how the platform can be intervened by the tools that will be later built by the IT Professional.

It is also important to consider that, depending on the landing page and organization size and complexity, many people can perform these two roles. As such, this method also considers possible that both roles are fulfilled by the same person.

On the other hand, the proposed method aims to establish continuous improvement in GUIs, and is composed by ten phases (shown in Fig. 1):

- (1) **Induction:** The goal of this phase is to identify the people involved in the continuous improvement project (among them, the main roles of “IT” and “UX”). In addition, landing page scope, its audience, its mission, its goals and its indicators should be identified.
- (2) **Intervention Elements Selection:** This phase aims to identify the elements that will be intervened by the method, following established criteria and business restrictions.
- (3) **Configuration:** This phase is utterly important, since it combines UX disciplines with those of Information Systems Engineering. With the information previously collected, the elements of the phase 2 are linked with their representation in the

source code. The method architecture and the parameters involved in the next phases are also defined in this phase.

- (4) **Initial Population Generation:** In this phase, the chromosome structure is defined, according to the project characteristics. In addition, the first generation chromosomes are created.
- (5) **GUIs modification:** This is the phase where the actual modifications take place. The phase describes how to create and intervene different landing page versions, according to the previously chosen architectural alternative.
- (6) **Deployment:** The goal of this phase is to make the intervened GUIs be available to the audience, as well as their related measurement tools.
- (7) **Measurement:** This phase remains for a defined amount of time with the objective of measuring user interactions, considering the audience type and indicators identified in phase 1. Pre-defined alerts can also occur to indicate anomalies in the method performance.
- (8) **Evaluation:** Once the previous phase has concluded, performance scores are assigned to the intervened interfaces, using a Fitness Function.
- (9) **Crossover and Mutation:** The goal of this phase is to obtain a new generation (that is, a new group of GUI variations) that own shared visual characteristics from previous GUIs with good performance. Furthermore, this phase adds new modifications to the new group of GUIs, in order to explore the effect of new perception phenomena on the audience.
- (10) **Closing:** When the termination criteria are met, the resulting GUI changes should be documented and analyzed. The resulting GUI is then permanently set as the definitive landing page.

As it can be seen in Fig. 1, and as this is a Continuous Improvement Method, Crossover and Mutation leads to the GUIs Modification in a cyclic manner. Alternatively, the method provides termination criteria that could lead to the Closing phase.

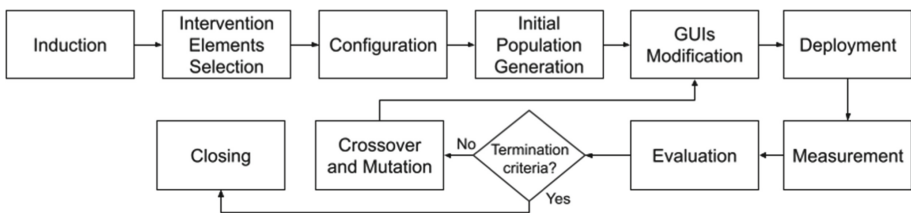


Fig. 1. Continuous improvement method for GUIs

3 Results

For the validation of the proposed method, a study case of a language services company is used [38]. In this case, the landing page plays a fundamental role in the client acquisition. The company is 7 years old and provides translation services, language conversational sessions and language online classes.

Although the company relies on online campaigns that are considered effective, with an established landing page that provides them with an acceptable Conversion Rate (measured with a value of 5.15% on January 10, 2021), the company is expecting to acquire a large amount of language service facilitators. For this reason, it is desired to increase the landing page Conversion Rate (leads generated divided by total sessions), as well as reduce the Bounce Rate (visitors without interaction, divided by total sessions). To achieve these goals, the company technology team will apply the proposed method to their landing page.

In the following sections, the application of each phase of the method is presented. Please note that during the results description, the company data is deliberately omitted in order to preserve the organization identity.

3.1 Induction

This phase is performed by IT and UX. The landing page is navigated entirely and, apart from the main page, five secondary pages are identified. It is decided that the intervention will occur on the main page only, documenting this decision in the induction record. In this case, the goals for the landing page are:

- Provide a better online experience, capturing the audience attention and potential customers. Its associated indicator is a Bounce Rate, expecting a value of less than 65% in the next 2 months.
- Attract new customers through the landing page. Its associated indicator is the action buttons Conversion Rate, expecting to reach a sustained value of 10% in the next 2 months.

3.2 Intervention Elements Selection

This phase includes three activities. In the first activity, the elements that will be intervened are selected, based on the landing page style guide, stored in the source code as a “.scss” file. So, no changes are made in the style guide. A visual representation of the style guide can be seen in the Fig. 2.

In the second activity, style alternatives are defined by UX, namely:

- *Typography*: On the current Landing Page all elements use the font “Roboto”, and it is planned to test “Lato” and “Exo” as font alternatives.
- *Color palette*: Currently the color palette is inspired by the company brand manual, and consists of the main color “Ocean Green” (# 44986E), the secondary color “Indian Red” (# D25A5A) and the tertiary color “Desert Storm” (# F2EBE2) of the landing page (see Fig. 2). UX proposes 54 color palette alternatives, and HSL distances are calculated for each color palette with respect to the original palette. The thirty-four most suitable palettes (closest distance to original palette) will be used in the next phases.



Fig. 2. Original landing page (left) and visual representation of the style guide (right)

In the last activity, the permutation groups are established, where UX proceeds to identify the landing page structural composition, as shown in Fig. 3. The most prominent visual restrictions identified by UX are:

- The page main sections (with the most visual impact and influence on the conversion rate) are Sect. 1 and Sect. 2. These sections can only be swapped with each other.
- Subsequent Sects. (3, 4, and 5) can be swapped with each other, with the only restriction of keeping the background color interleaved, to facilitate the distinction between sections.
- Sect. 6 (footer) should always be kept as the last section.

Considering these restrictions and the elements visual hierarchy, UX decides to create four permutation groups (Fig. 3):

- Permutation group composed with the main sections: “Sect. 1” and “Sect. 2”.
- Permutation group with the secondary landing page sections: “Sect. 3”, “Sect. 4” and “Sect. 5”.
- Permutation group composed with the Carousel elements: “Carousel 3 images”.



Fig. 3. Landing page structural composition

3.3 Configuration

The four activities corresponding to this phase are: (I) Architecture evaluation, (II) Obtaining and integrating the analytics tool, (III) Linking the Intervention Elements with their representation in source code and, finally, (IV) Parameterization.

In activity I, IT reviews the criteria established by the method and decides to use the offline architecture (this means, automatically modifying source code by parsing instead of editing the DOM).

Later in Activity II is expected to get a tool that fetches the landing page Conversion Rate and Bounce Rate. Then, Google Analytics is integrated onto the existing landing page, and each chromosome will have a unique tracking ID. After the analytics integration, a test is carried out in a period of three days, obtaining an initial bounce rate of 73%, conversion rate of 5.15%, sessions per user of 1.19, and session duration of 17 s with 145 evaluated sessions in total.

In Activity III, the Permutation Elements (Fig. 3) are spotted in the source code and labeled through HTML class names.

Finally, in Activity IV, IT and UX determine the parameters that will be used as input information for the method tools:

- *Individuals per generation*: Based on the available tracking IDs and the sessions evaluated in activity II and, the method will create 10 individuals per generation. As indicated in [34], this population size is considered small enough to gather many sessions per chromosome, while big enough to achieve good results.
- *Fitness criteria*: Based on the information stated in phase 1, Conversion Rate represents the 70% of the fitness function, and Bounce Rate represents the remaining 30%.
- *Generation duration*: Based on the number of sessions evaluated, it is determined that seven days is a suitable generation duration.
- *Audience reach for new and old individuals*: 20% of a previous generation (the 2 best individuals) will remain unchanged in the next generation. Then 8 new individuals will be created.
- *Termination criteria*: The method will reach its closing phase when the best individual of a generation reaches a Conversion Rate of 10% and a Bounce Rate of 65%.

3.4 Initial Population Generation

The three activities in this phase are: (I) Original chromosome description, (II) Chromosomes creation, and (III) Non-interactive fitness function: discard invalid candidates.

Firstly, in activity I, IT describes the chromosome structure (Table 1). Next, IT develops and implements micro services to create and modify these structures.

Secondly, in Activity II, the chromosomes that would be part of the first generation are created. This process is based on the registered permutation groups, and style elements that are chosen randomly. One tracking ID is obtained for each chromosome.

Finally, in Activity III, a function is developed to discard chromosomes whose fitness is so low that it compromises the landing page visual experience. This is based on the difference in luminosity units between colors in the palette (comparing primary, secondary and tertiary colors). Thus, low-contrast, visually conflicting chromosomes are replaced with new randomly (and valid) generated individuals.

3.5 GUIs Modification

The two activities in this phase are: (I) Architecture Replication, and (II) Source code intervention.

First, the services used to host the landing page are replicated ten times (one for each landing page) and an extra service is created to randomly redirect a new user to any of these individuals.

Finally, a toolkit that automatizes workflow is used, since it was already applied in the project. Automated tasks are created to modify the source code. CSS code is added to reflect the chromosomes style elements and color palette, and HTML elements are swapped to reflect the permutation groups.

Table 1. Chromosome structure

Field name	Data type
ID	INTEGER
perm_group_1_carousel	STRING[]
perm_group_2_main_sections	STRING[]
perm_group_2_secondary_sections	STRING[]
main_color_H	INTEGER
main_color_S	INTEGER
main_color_L	INTEGER
secondary_color_H	INTEGER
secondary_color_S	INTEGER
secondary_color_L	INTEGER
tertiary_color_H	INTEGER
tertiary_color_S	INTEGER
tertiary_color_L	INTEGER
font_family	STRING
generation	INTEGER
times_requested	INTEGER
analytics_tracking_id	INTEGER
bounce_rate	FLOAT
sessions_per_user	FLOAT
session_duration	FLOAT
bounce_rate	FLOAT
total_score	FLOAT

3.6 Deployment

IT works with the tool built in the previous phase to add the capacity to automatically deploy the individuals in each of the replicated hostings. A cookie is added and read in the user device to make sure they are redirected to the same individual, and thus assuring visual consistency among sessions.

3.7 Measurement

In this phase, two activities are performed: (I) Obtaining results, (II) Results recording.

First, in Activity I, the resources previously configured are used to create a tool that obtains the performance metrics for each chromosome. Then, in Activity II, after the seven days of the first generation, the performance results are obtained.

3.8 Evaluation

The three activities belonging to this phase are: (I) Determine and calculate the fitness function, (II) Evaluation of the termination criteria, and (III) Individuals selection.

In Activity I, the fitness function is defined as in Eq. 1:

$$F(X) = \frac{(A - R0a)}{(R1a - R0a)} * \alpha + \frac{(B - R0b)}{(R1b - R0b)} * \beta \quad (1)$$

X is an individual to be evaluated by the fitness function. For the performance dimension “a” (A), which refers to the Conversion Rate, R0a and R1a are their lower and upper performance limits respectively, and α is the fitness criteria defined in Configuration. This shows the measured performance of dimension “a” (A), the lower and upper performance limit of dimension “a” (R0a and R1a, respectively), and the coefficient defined by the aptitude criterion in phase configuration, activity IV. The same rationale applies to dimension “b” (Bounce Rate), and respective limits (R0b and R1b) and weight (β).

To set lower performance limits, the first-generation lowest measurements are considered, while the upper performance limits are set by the values stated in the landing page goals in phase 1 (Table 2).

Table 2. Lower and upper performance limits.

	Conversion rate	Bounce rate
Lower performance limit	0.03	0.879
Upper performance limit	0.10	0.65

Therefore, the generic fitness function defined in Eq. 1 becomes:

$$F(X) = \frac{(A - 0.03)}{(0.10 - 0.03)} * 0.7 + \frac{(B - 0.879)}{(0.65 - 0.879)} * 0.3 \quad (2)$$

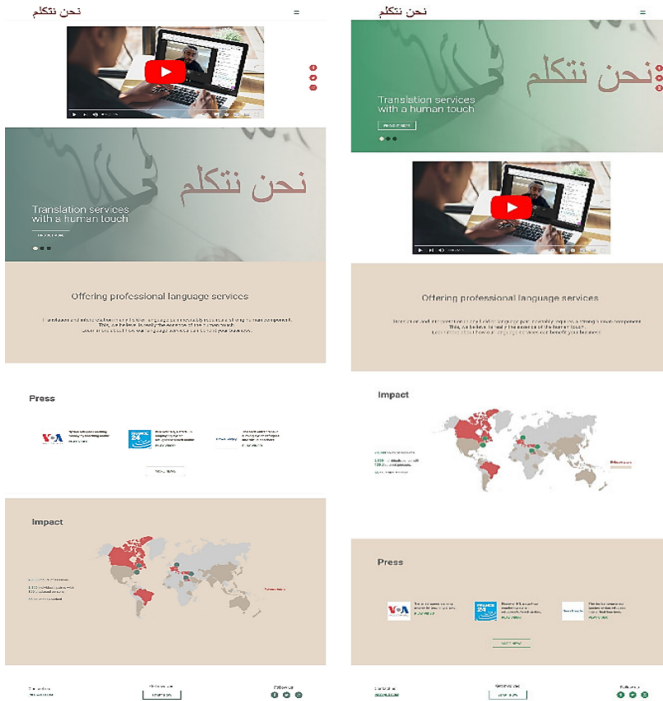


Fig. 4. Best performing landing page (left) and worst performing landing page (right) from the first generation

The Activity II confirms that the best individual performance does not meet the termination criteria established in the Configuration, presenting a Conversion Rate of 0.061 and a Bounce Rate of 0.686, and more generations are needed. The best and worst performing landing pages from the first generation can be seen in Fig. 4.

In Activity III, first generation individuals are chosen by the “ranking” selection method. The Chromosome Creator tool developed in the phase 4 assign copies to the next generation: The three worst performing chromosomes are assigned with 0 copies, whilst the three best performing chromosomes are assigned with 2 copies.

3.9 Crossover and Mutation

In this phase, three activities are carried out: (I) Crossover, (II) Mutation and (III) Inclusion of the null hypothesis (if needed).

First, in activity I, IT works on the Chromosome Creator tool to generate new chromosomes. As previously defined, two copies belonging to the best performing individuals remaining as they are. Then, random binomial crossover is performed in the rest of the copies, creating chromosomes with the shared characteristics of their parents combined.

Then, in Activity II, the chosen chromosomes for the mutation experience an alteration in one randomly chosen gene. The gene value is replaced in the same way genes were created in phase 4.

Finally, activity III is not performed until the creation of the third generation.

3.10 Subsequent Generations

In this section, the subsequent generations (created in each iteration of the method) are analyzed. At this point, the method is running completely automated.

The second-generation starts with the GUIs modification phase and, after the deployment, the individuals' performance is measured trough 7 days. As a result, in this second generation, the best performing chromosome does not meet the termination criteria, counting with a Conversion Rate of 0.0882 and a Bounce Rate of 0.588. This means that more generations are needed. The best and worst performing landing pages from the second generation are shown in Fig. 5.

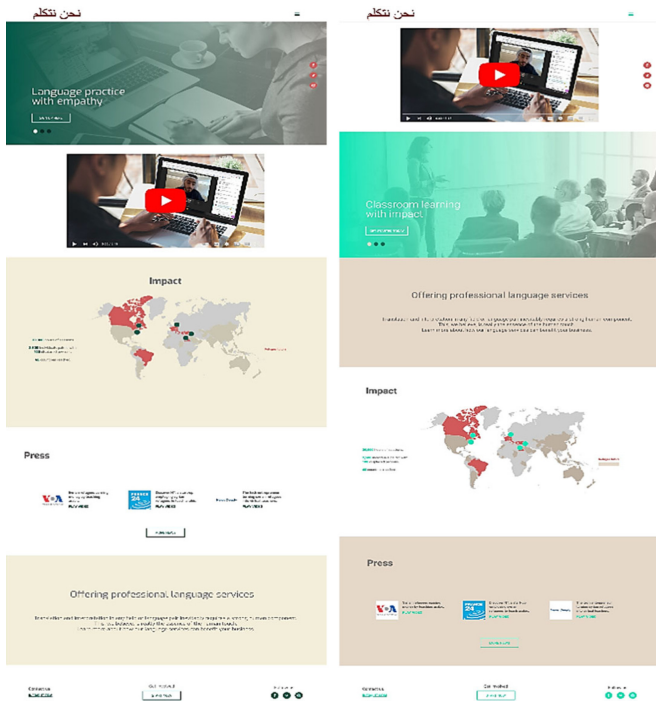


Fig. 5. Best performing landing page (left) and worst performing landing page (right) from the second generation

Therefore, the third generation is created with ten new chromosomes, and the method iterates again. Once again, the best performing chromosome does not meet the termination criteria, showing a Conversion Rate of 0.0968 and a Bounce Rate of 0.6129. In addition, in this generation, null hypothesis should be included. After replacing a chromosome for another already existing one, similar metrics are found in both cases, with the average of the sessions duration varying only slightly (11.28 vs 13.57). The best and worst pages from this generation are shown in Fig. 6.

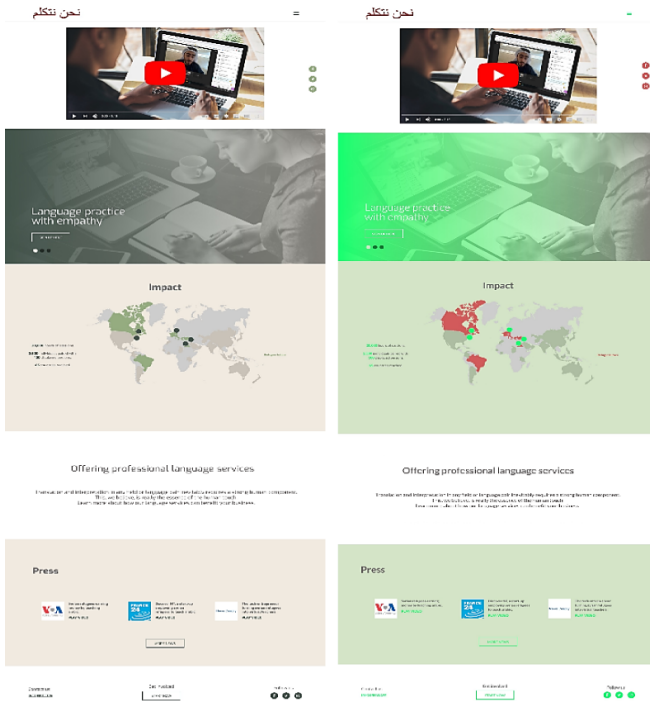


Fig. 6. Best performing landing page (left) and worst performing landing page (right) from the third generation

In the fourth generation, the method is iterated once again, but this time the best performing chromosome meets the termination criteria, having a Conversion Rate of 0.11 and a Bounce Rate of 0.615. Figure 7 shows the best and worst performing landing pages from this fourth generation.

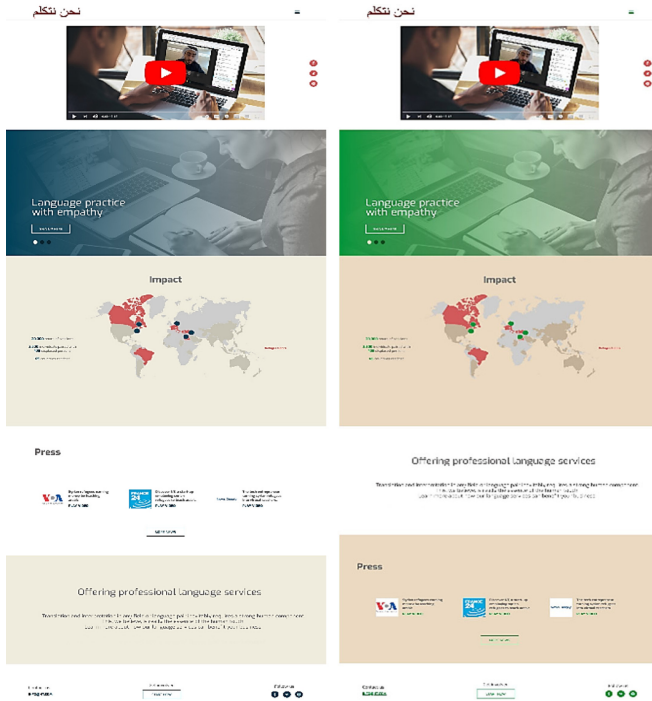


Fig. 7. Best performing landing page (left) and worst performing landing page (right) from the fourth generation

Although the method should continue with the Closing phase, two more generations are performed to obtain a more profound method analysis. In the fifth generation, ten new chromosomes are exposed. The corresponding performing landing pages from the fifth generation are shown in Fig. 8.

In addition, the sixth generation is performed to evaluate the performance of the most successful chromosome in the most successful generation. For this, chromosome 40 (in fourth generation) is selected, and ten chromosomes with the same characteristics are created. The results of these generations will be addressed later in Sect. 4.

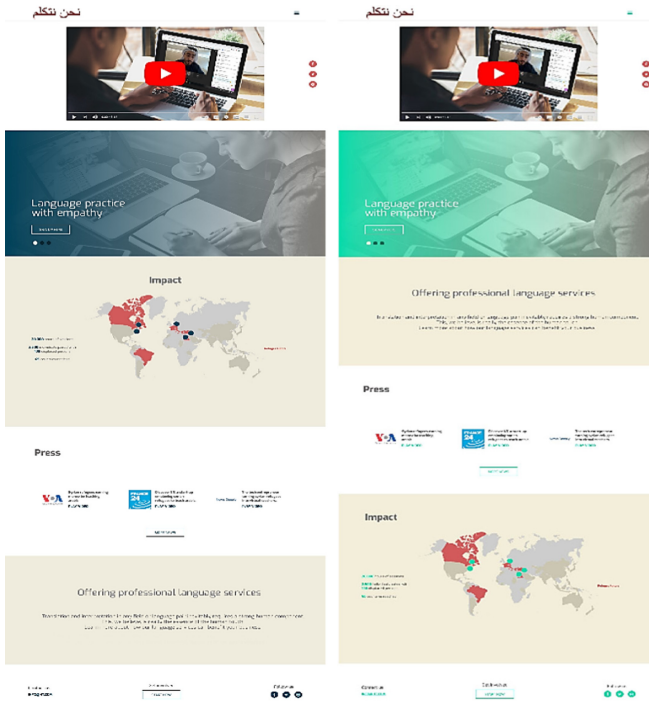


Fig. 8. Best performing landing page (left) and worst performing landing page (right) from the fifth generation

3.11 Closing

After six generations, the closing activities are performed by IT and UX. The landing page is presented to the project stakeholders, showing the changes introduced by the method tools and their impact on the landing page performance. The page final version is shown in Fig. 9 where it can be seen important changes compared to the initial version (available in Fig. 2). The changes proposed by chromosome 40 are reviewed, and the landing page’s source code is updated to match these changes.

The landing page is then deployed and all services provided by the method tools are paused. The source code of these tools is stored in a repository, in case the method is implemented in the future, and all the chromosome records are stored in a document. Location and credentials to access this information is stated in the “Closing document”.



Fig. 9. Final landing page (Chromosome 40)

4 Analysis of the Results

For two and a half months, the Continuous Improvement Method for GUIs was deployed. By doing this, after the fourth iteration, the landing page of a language services company has reached the goals for which it has been designed. The performances of the fifty generated landing pages along the six generations are shown in Fig. 10.

Reviewing these results by comparing the initial and final version, some improvements can be noticed:

- Average Conversion Rate has increased from 0.0515 to a sustained value between 0.1007 and 0.102.
- Average Bounce Rate, a secondary performance dimension, has decreased from 0.78 to a sustained value between 0.60 and 0.63.
- Other performance dimensions not considered in the landing page goals are shown in Fig. 10: Average Sessions per User have increased from 1.19 to 1.367 and Average Session Duration has increased from 17 s to 27.5 s. This emphasizes the holistic nature of the perception phenomena, where Conversion Rate or Bounce Rate improvements are also reflected in the overall user experience.

The null hypothesis included in the third generation reflects that the method experiments are consistent: two chromosomes sharing the same characteristics have similar performance. This is also reflected in the sixth generation.

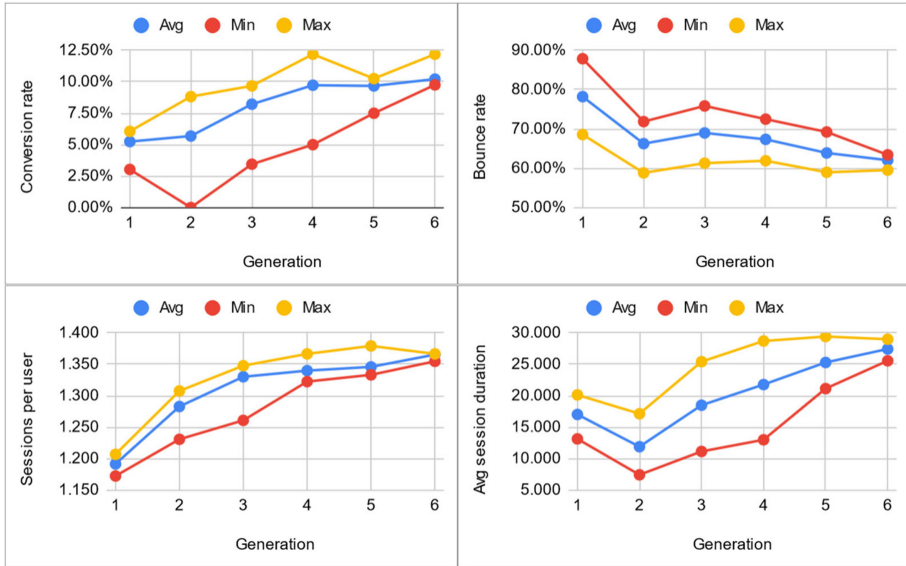


Fig. 10. GUIs performance along six generations

An interesting analysis can be done by observing the fifty generated GUIs. Some visual characteristics have gradually populated the generations, enabling their convergence (i.e. squared font families). Another visual example is the order of images in the carousel, where “conversational sessions” have been preferred over “translate services”.

In addition, it has been determined that landing pages that included the promotional video at the top presented better performance, even when this meant that the action buttons were not easily available. This reinforces the previously stated concept that GUI design is a complex (and even counter intuitive) domain that could be improved through experimentation, automation and intelligent systems.

5 Conclusions

In this paper it is determined that the Continuous Improvement Method for GUIs can be applied with positive results, helping GUIs to achieve the goals for which they were designed. Therefore, this method assists individuals among the computer science and user experience disciplines who are involved in the design, development and implementation of web based GUIs. As a result of applying a semi-automatic method, using AGIs and feedback from end users, it establishes continuous improvement, in order to ensure low costs in development and implementation, while maximizing GUI performance.

Based on the above, further analysis and validations of the method could be carried out in other contexts. This could lead to the identification of common rules or patterns that govern the method evolution through its generations (such as the impact of different levels of exploration in performance or the impact of different selection, mutation or crossover techniques).

On the other hand, and based on the performance results of the method application in this and other cases, Artificial Intelligence algorithms could be implemented (such as ID3) to obtain hypotheses that relate the presence or absence of visual characteristics with improvements in the GUIs performance dimensions.

Finally, it is recommended to experiment with more complex chromosomes. This means, adding new Intervention Elements (besides permutation and style elements) or even adding new possible values to the aforementioned elements, to widen the possibilities of visual modification.

References

1. Miller, S., Huber, S.: *A Bíblia e Sua História: O Surgimento e o Impacto da Bíblia*. Sociedade Bíblica do Brasil, Rio de Janeiro (2017)
2. Passos, J.E., Silva, T.L.: *La Evolución Tecnológica y su Impacto en el Diseño de la Interfaz: Diplomado en Diseño de la Universidad Federal de Rio Grande do Sul* (2013). <http://www.bocc.ubi.pt/pag/passos-silva-2013-la-evolucion-tecnologica-impacto.pdf>
3. Santaella, L.: *Navegar no ciberespaço: O perfil cognitivo do leitor imersivo*. Paulus, São Paulo (2004)
4. Albornoz, M.C., Berón, M., Montejano, G.: *Interfaz gráfica de usuario: el usuario como protagonista del diseño*. In: XIX Workshop de Investigadores en Ciencias de la Computación (WICC 2017, ITBA, Buenos Aires), pp. 570–574 (2017)
5. Ramírez, D., Cervantes, J., Molina, J., López, S.: *Requirements engineering to design a brain-computer interface for cognitive training*. In: 2018 7th International Conference On Software Process Improvement (CIMPS), pp. 79–84 (2018)
6. Moshagen, M., Tielsch, M.T.: *Facets of visual aesthetics*. *Int. J. Hum.-Comput. Stud.* **68**, 689–709 (2010)
7. Sbaffi, L., Rowley, J.: *Trust and credibility in web-based health information: a review and agenda for future research*. *J. Med. Internet Res.* **19**(6), e218 (2017)
8. Ponte, E.B., Carvajal-Trujillo, E., Escobar-Rodríguez, T.: *Influence of trust and perceived value on the intention to purchase travel online: integrating the effects of assurance on trust antecedents*. *Tour. Manag.* **47**, 286–302 (2015)
9. Hasan, B.: *Perceived irritation in online shopping: the impact of website design characteristics*. *Comput. Hum. Behav.* **54**, 224–230 (2016)
10. Tielsch, M.T., Blotenberg, I., Jaron, R.: *User evaluation of websites: from first impression to recommendation*. *Interact. Comput.* **26**(1), 89–102 (2014)
11. *Watermark Consulting: The Watermark Consulting 2013 Customer Experience ROI Study* (2013). <https://www.watermarkconsult.net/blog/2019/01/14/customer-experience-roi-study/>
12. Sánchez, W.: *La usabilidad en Ingeniería de Software: definición y características*. *Ingeniería* **1**(2), 7–21 (2011)
13. Pina, P.: *What's perfect for me? How CPG shoppers are making the best choice* (2018). <https://www.thinkwithgoogle.com/consumer-insights/consumer-packaged-goods-industry/>
14. Lee, A.: *The Future of Design in Start-Ups Survey: 2016 Results* (2016). <https://www.nea.com/blog/the-future-of-design-in-start-ups-survey-2016-results>

15. Kohavi, R., Longbotham, R.: Online controlled experiments and A/B testing. *Encyclopedia Mach. Learn. Data Min.* **7**(8), 922–929 (2017)
16. Kohavi, R., Crook, T., Longbotham, R.: Online experimentation at Microsoft (2009). <https://exp-platform.com/experiments-at-microsoft/>
17. Agung Zulkarnaen, A.: Pengembangan Web Landing Page CMS Wordpress Dengan Plugin Thrive Architect Pada Acara Entreprenuer Gathering 5. Technical report, studentstaff5, Perpustakaan Institut Teknologi Telkom Purwokerto (Unpublished) (2018)
18. Gafni, R., Dvir, N.: How content volume on landing pages influences consumer behavior: empirical evidence. In: *Proceedings of the Informing Science and Information Technology Education Conference*, La Verne, California, pp. 035–053 (2018)
19. Forsythe, A., Nadal, M., Sheehy, N., Cela-Conde, C.J., Sawey, M.: Predicting beauty: fractal dimension and visual complexity in art. *Br. J. Psychol.* **102**(1), 49–70 (2011)
20. Michailidou, E., Harper, S., Bechhofer, S.: Visual complexity and aesthetic perception of web pages. In: *Proceedings of the 26th Annual ACM International Conference on Design of Communication*, pp. 215–224 (2008)
21. Bringas Hidalgo, A.: *Psicología*. Esfinge, Mexico (2010)
22. Seckler, M., Opwis, K., Tuch, A.N.: Linking objective design factors with subjective aesthetics: an experimental study on how structure and color of websites affect the facets of users' visual aesthetic perception. *Comput. Hum. Behav.* **49**, 375–389 (2015)
23. Cyr, D., Head, M., Larios, H.: Colour appeal in website design (within and across cultures: a multi-method evaluation. *Int. J. Hum. Comput. Stud.* **68**(1–2), 1–21 (2010)
24. Cyr, D., Head, M.: Website design in an international context: the role of gender in masculine versus feminine oriented countries. *Comput. Hum. Behav.* **29**(4), 1358–1367 (2013)
25. Sharples, M.: Human-computer interaction. In: Boden, M. (ed.) *Artificial Intelligence*, pp. 293–323. Oxford University Press, Oxford (1996)
26. Kolko, J.: Design thinking comes of age (2015). https://cdn.fedweb.org/fed-42/2892/design_thinking_comes_of_age.pdf
27. Srinivas, M., Patnaik, L.M.: Genetic algorithms: a survey. *Computer* **27**(6), 17–26 (1994)
28. Mirjalili, S.: *Evolutionary Algorithms and Neural Networks*. SCI, vol. 780. Springer, Cham (2019). <https://doi.org/10.1007/978-3-319-93025-1>
29. Cho, S.B.: Towards creative evolutionary systems with interactive genetic algorithm. *Appl. Intell.* **16**(2), 129–138 (2002)
30. Miikkulainen, R., et al.: Conversion rate optimization through evolutionary computation. In: *Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1193–1199 (2017)
31. Blanchard, B.S.: *Ingeniería de sistemas*. Isdefe, Madrid (1995)
32. Sommerville, I.: *Ingeniería del Software*. Pearson educación, Madrid (2005)
33. Macaulay, L.A.: *Requirements Engineering*. Springer, London (2012)
34. Quiroz, J.C., Louis, S.J., Shankar, A., Dascalu, S.M.: Interactive genetic algorithms for user interface design. In: *2007 IEEE Congress on Evolutionary Computation*, pp. 1366–1373 (2007)
35. Florez, H., Garcia, E., Muñoz, D.: Automatic code generation system for transactional web applications. In: Misra, S., et al. (eds.) *ICCSA 2019. LNCS*, vol. 11623, pp. 436–451. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24308-1_36
36. Ndadji, M.M.Z., Tchendji, M.T., Djamegni, C.T., Parigot, D.: A language and methodology based on scenarios, grammars and views, for administrative business processes modelling. *ParadigmPlus* **1**(3), 1–22 (2020)
37. Mendez, O., Florez, H.: Applying the flipped classroom model using a VLE for foreign languages learning. In: Florez, H., Diaz, C., Chavarriaga, J. (eds.) *ICAI 2018. CCIS*, vol. 942, pp. 215–227. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01535-0_16
38. Gómez, P., Sánchez, M.E., Florez, H., Villalobos, J.: An approach to the co-creation of models and metamodels in enterprise architecture projects. *J. Object Technol.* **13**(3), 1–29 (2014)