

ゲーミフィケーションのルールと個人の嗜好がソフトウェア開発に及ぼす影響

吉上 康平^{†1} 林 大志^{†2} 角田 雅照^{†2,a)} 上野 秀剛^{†3,b)}
佐々木 俊一郎^{†2} 松本 健一^{†1}

ソフトウェア工学の教育や実際のソフトウェア開発に対して、作業の成果などを高めるために、ゲーミフィケーションを適用することが試みられている。ゲーミフィケーションとは、ゲームの要素をゲーム以外のコンテキストに導入することを指す。ゲーミフィケーションを適用する場合、作業時間に制限を設けるなど、様々なルールが考えられる。ただし、そのルールが作業時間などにどのような影響を与えるかは明らかでない。また、個人の嗜好などにより、ルールの効果が異なる可能性がある。例えば性急な性格を持つ作業者に対し、作業時間を制限するルールを適用すると、作業時間を短縮する可能性もあれば、焦りから作業品質に悪影響を与える可能性もある。そこで本研究では、ルールを変化させると作業時間などに違いが生じるのか、ゲーミフィケーションの効果は個人の嗜好の差異に影響されるのかを実験により分析した。その結果、コーディングに関しては、ルールにより作業時間が短縮される可能性があることが示された。また、個人の嗜好の差異は、ゲーミフィケーションの効果にあまり影響していないことが示された。

キーワード： 人的要因, モチベーション, 作業効率

1. はじめに

ソフトウェア開発において、作業の効率を高めることは非常に重要な課題のひとつである。近年、そのためのアプローチとして、ゲーミフィケーションをソフトウェア開発関連の活動に適用することが試みられている。ゲーミフィケーションとは、「ゲームの要素をゲーム以外のコンテキストに導入すること」[4]と定義されており、作業意欲や成果を高めることを狙いとしている。ゲーミフィケーションでよく用いられる要素として、他者との競争や、架空のトロフィーやバッジの収集などがあげられ、これらにより作業者のモチベーションを高めようとしている。

ソフトウェア開発に関連するゲーミフィケーションの適用例として、例えばソフトウェア工学関連の学習や訓練などがあげられる[1][5]。いくつかの研究では、実際のソフトウェア開発に適用することを前提としており[19]、実際の開発に適用した事例もある[15]。

ただし、ソフトウェア開発におけるゲーミフィケーションの効果は、まだ十分に評価されているとはいえない。例えば、ソフトウェア開発関連のゲーミフィケーションの主要な研究では、約半数が実験による評価を行っていない[17]。また、定量的な評価を行っている場合でも、作業時間を評価している研究は我々の知る限り多くない。

本研究では、ソフトウェア開発に関連する活動に対し、ゲーミフィケーションを適用する場合、その効果をより高めるために、以下を明らかにしておくべきであると考え。

- ゲーミフィケーションのルールを変化させると、作業時間などに違いが生じるのか?
- ゲーミフィケーションの効果は、個人の嗜好の差異に影響されるのか?

ゲーミフィケーションをソフトウェア開発に適用する場合、様々なルール設定が考えられるが、そのルールが作業時間などに影響する可能性がある。例えば、ゲーミフィケーションのルールとして、作業時間に制限を設けた場合、作業時間が短縮される可能性があるが、逆に心理的に焦りが生じることにより、作業の成果（品質）に何らかの悪影響を及ぼす可能性がある。具体的には、コードレビューにおいて時間制限のルールを適用した場合、レビュー時間は短くなる可能性があるが、逆にレビュー指摘数（発見欠陥数）が低下する可能性がある。

作業時間と作業の成果には一定のトレードオフが存在すると考えられ、作業時間を短くすればするほどよいというものではない。一方で、一定の品質が保たれていれば、作業時間をできるだけ短くして納期などに間に合わせるほうが重要な場面もありうる。そこで本研究では、ルールにより作業時間が短縮されるのかどうかを中心に分析を行う。ゲーミフィケーションのルールの影響を明らかにすることにより、どのようにルールを設定すべきかの指針となると考える。

また、個人の嗜好などにより、ゲーミフィケーションの効果が異なる可能性がある。例えば、単純ではあるが、ゲーミフィケーションを好むほど、作業時間が短縮されるなどの効果が大きい可能性がある。さらに、個人の性質もゲーミフィケーションの効果と関連がある可能性がある。例えば、焦りが生じやすい性質を持つ開発者に対し、時間制限のルールを適用した場合、逆に心理的プレッシャーから作業効率が低下することもありうる。本研究では、個人の

†1 奈良先端科学技術大学院大学
Nara Institute of Science and Technology, Japan

†2 近畿大学
Kindai University, Japan

†3 奈良工業高等専門学校
National Institute of Technology, Nara College, Japan

a) tsunoda@info.kindai.ac.jp

b) uwano@info.nara-k.ac.jp

性格を把握するための指標として、行動経済学で定義されている時間割引率[7]を用いる。個人の嗜好を考慮することにより、ゲーミフィケーションの各種ルールの効果をより高められる可能性がある。

本研究では、ゲーミフィケーションのルールの差異が作業成果に影響するかどうか、また、ゲーミフィケーションの結果が個人の嗜好などに影響されるかどうかを分析する。そのために、実験においてコードレビューとコーディングの2つのタスクを被験者に与え、その結果を分析した。

2. ゲーミフィケーション

ゲーミフィケーションとは、「ゲームの要素をゲーム以外のコンテキストに追加すること」[4]であり、作業に娯楽性を付加することによる作業意欲の向上を狙いとしている。ソフトウェア開発に限らず、近年非常に注目されているアプローチである[13][22]。ゲーミフィケーションの目的は、作業効率や持続性の向上である。例えばゲーミフィケーションの要素として以下のような要素がある[9][17]。

- ポイント: 何らかの作業を終えるとポイントが付与される。
- ランキング: ポイントの上位者が他の参加者に示される。
- バッジ: 何らかの作業を終えると、作業に応じたバッジが(システム上で)入手できる。
- レベル: 一定のポイントに達するごとにレベルが上がる。

ある作業を完了した際に何らかのアイテムやバッジなどが入手できるようにすると、それらを収集したいという欲求を刺激し、作業に対するモチベーションが高まることが期待される。また、作業に対して得点(スコア)を算出すると、過去の自分、もしくは他人との競争が可能になり、モチベーションが高まることが期待される。

Pedreira ら[17]は、ソフトウェア工学におけるゲーミフィケーションの研究をマッピングしている。具体的には、ターゲットとしている開発プロセス、用いているゲーミフィケーションの要素、ゲーミフィケーションの効果の評価方法に基づいてマッピングしている。ここで取り上げられた研究のうち、約半数は実験による評価を行っておらず、手法の提案のみにとどまっているものも多い。

ゲーミフィケーションを教育や学習に適用した事例は比較的多い。例えば Akpolat ら[1]は、エクストリームプログラミングの学習にゲーミフィケーションを導入し、その効果を学生にアンケートすることにより確かめている。Dubois ら[6]は、実際のソフトウェア開発にゲーミフィケーションを導入することを最終的なゴールとしつつ、最初のステップとして教育において導入している。学部生が卒業制作としてソフトウェアを作成する際にゲーミフィケーションを導入することにより、テストカバレッジなどいくつか

の項目が改善したことを示している。Singer[19]も、実際のソフトウェア開発にゲーミフィケーションを導入することをゴールとし、バージョンコントロールシステムにゲーミフィケーションを導入する(一定数以上コミットするとバッジがもらえる、各メンバーのコミット数などが通知される)ことを提案している。実験は学生に対して適用し、評価は学生に対するアンケートを定性的に行っている。Souza ら[21]は、ソフトウェア工学の学習におけるゲームの活用事例について調査している。Souza らは、学習におけるゲームを、ゲーミフィケーション、本物のゲーム(パズルゲームなど)、ゲーム開発の3種類に分類し、後者2つを用いた研究が多いことを示している。

教育への適用に限定していない研究もいくつか存在する。Arai ら[2]は、FindBugsなどの警告を、ユーザが積極的に除去するように、ゲーミフィケーションを導入することを提案している。取り除かれた警告の割合を、ゲーミフィケーションの有無で場合分けして比較することにより、その効果を確かめている。Smith ら[20]は、ソフトウェアの品質に関する調査への参加意欲を高めるために、ゲーミフィケーションを用いている。

ゲーミフィケーションを企業へ適用する試みもいくつか行われている。例えば、Stanculescu ら[22]は、ゲーミフィケーションを企業内に適用し、SNSでの振る舞い(ニュースシェアの数などに基づく評価)と学習(アンケートで評価)に効果があることを示している。根本[15]は、実際のソフトウェア開発においてゲーミフィケーションを導入し、テストにおいてより多く欠陥を発見できたことを示している。StackOverflowはゲーミフィケーションの要素を導入しているといえるが、StackOverflowに関して分析した研究もいくつか存在する[26][27]。例えば Vasilescu ら[27]は、StackOverflowとGitHubのアクティビティの関連について分析し、StackOverflowの回答者は実際の開発者であり、時間を持て余している人物が、バッジのためだけに活動しているのではないと結論づけている。

ソフトウェアのレビューに対してゲーミフィケーションの導入を試みた研究がいくつか存在する。例えば Unkelos-Shpigel ら[24]は、ゲーミフィケーションに基づくコードレビューのツールを提案している。ただし、評価は一切行っていない。ゲーミフィケーションの評価では、累積の作業成果(欠陥の発見数やコミットの数)に基づくものは多いが、作業時間も考慮して評価している研究は、我々の知る限り多くない。Khandelwal ら[14]は、コードレビューにおいてゲーミフィケーションを導入し、その有無による作業時間の差を分析している。ただし、ゲーミフィケーションのルールが変わると、作業結果にどのような影響が出るのか(ルールを工夫すると作業時間を短縮することができるのか)、さらに、個人の性質とゲーミフィケーションのルールとの関係は、我々の知る限り定量的な評価

は行われていない。

3. 時間割引率

3.1 定義

時間割引率は行動経済学で定義されている概念である。行動経済学とは、人が必ずしも合理的に活動しないことを考慮して経済活動を分析する学問である。経済学では、将来の金銭的価値を現在に置き換えて評価する場合には、将来の価値を額面通り評価するのではなく、一定の比率だけ割り引いて評価すると想定している。標準的経済学で使用される指数型割引モデルでは、 t 期後の Y 円を現在評価するときの割引現在価値 X は以下の式(1)で評価される。

$$X = \frac{Y}{(1+r)^t} \quad (1)$$

r は時間に関する好みを表すパラメータであり、割引率と呼ばれる。

ここで割引率が個人間で異なると想定して数値例を考える。個人 A の割引率が $r=0.5$ である場合、彼は 1 年後の 100000 円の割引現在価値を式(2)のように評価する。

$$\frac{100000}{1.5^1} = 66667 \text{円} \quad (2)$$

また、個人 B の割引率が $r=0.1$ である場合、彼は 1 年後の 100000 円の割引現在価値を式(3)のように評価する。

$$\frac{100000}{1.1^1} = 90909 \text{円} \quad (3)$$

もし、両個人が今日 80000 円受け取ることと、1 年後に 100000 円受け取ることのどちらかを選択できるとすると、彼らは今日の 80000 円と 1 年後の 100000 円の割引現在価値を比較するので、個人 A は今日 80000 円を受け取ることを選び、個人 B は 1 年後に 100000 円受け取ることを選ぶだろう。このように、時間割引率は異時点間の意思決定における個人の時間に対する性急さを表すパラメータであり、その値が大きいほどせっかち(impatient)であると解釈できる。

3.2 分析での利用

本研究において、ゲーミフィケーションと関連する個人の性質を計測するために、時間割引率を用いる理由は主に以下の 2 つである。

- ゲーミフィケーションの結果を報酬とみなすと、時間割引率は報酬に対する個人の態度を示すため、分析に(比較的)適している。
- 時間割引率は性急さを表すと説明されるため、性急な人が影響を受ける可能性がある時間制限のルールなどの影響分析に適している。

ゲーミフィケーションの結果は、ポイントやバッジであり、これはある種の報酬であるとみなすことができる。ゲーミフィケーションの結果(報酬)はすぐに得られるため、例えば時間割引率の高い人ほど、ゲーミフィケーションを

表 1 利得表の例[12]

A (2 日後受取)	B (9 日後受取)	金利
100,000	100,019	1%
100,000	100,192	10%
100,000	100,288	15%

好む可能性がある。

また、時間割引率は性急さを表すと説明される。性急な人は、タスクの時間制限を強調したルールに反応しやすい可能性がある。例えば、時間割引率の高い人は、そうでない人よりも作業速度が速まる可能性もあり、逆に焦りが生じて作業効率が低下することも考えられる。さらに、性急な人は、根気強くコードを読む必要があるレビューなどの作業に向いていない可能性があり、その点に関しても分析が可能となる。

3.3 計測方法

時間割引率は、被験者に表 1 のような条件の場合、A と B のどちらかを選んでもらうことにより求めた。これは多くの研究で用いられている方法である[23]。被験者は、金利(9 日後に受け取る金額)が低い場合は A を選択するが、ある一定よりも金利が高くなると、それ以降は B を選ぶ。そこで、選択が切り替わった時点の金利の平均値を時間割引率とする。例えばある被験者が金利 10%までは A を選択し、15%の場合は B を選択した場合、選択が切り替わる前後の金利の 10%と 15%を平均して 12.5%を時間割引率とする。全ての選択が A の場合、金利は最大の 15%とする(全選択が B の場合も、同様に最小の 1%とする)。

A と B の日数の差などは文献[12]と同様に 4 パターンとした。例えばあるパターンでは、A は 2 日で B は 9 日後、A の金額は 10 万円、金利は-10%から 400%を 15 段階に変化させ、この組み合わせを変えて 4 パターンとしている。得られた 4 つの時間割引率の平均値を被験者の時間割引率とした。例えば、4 パターンの利得表から、時間割引率が 10%、5%、15%、10%となった場合、それらを平均した 10%を被験者の時間割引率とした。

4. 実験

実験では、被験者にレビュー、コーディングのタスクを順番に行ってもらい、コーディングのタスク直後にアンケートを実施した。前者ではコードの誤りを指摘してもらい、後者では仕様に従ってコーディングをしてもらった。レビューとコーディングのタスクの間は 1 週間以上空いている。被験者には、実験後に自分の順位を知らせる(ただし他の被験者には結果を開示しない)ことを事前に伝え、競争心が高まる工夫を行った。被験者は情報科学を専攻する学部生であり、レビューのタスクの被験者数は 13 人、コーディングのタスクは 14 人である。両方のタスクにおいて、被験者はコーディングのタスク 1 名を除いて同じである。

クラス	行	変更点
sum	7	i--ではなくi++
main	13	mではなくn
sum	3	sum=1ではなくsum=0

得点 問題1	指摘1件あたり
600	200

(a) ルール α の場合

クラス	行	変更点
sum	7	i--ではなくi++
main	13	mではなくn
sum	3	sum=1ではなくsum=0

得点 問題1	平均	指摘1件あたり
1168	1000	200

(b) ルール β の場合

図 1 レビュータスク時の画面イメージ

なお、いくつかの研究[10][18]において、学生と実際の開発者の両者を被験者として実験した場合、被験者による結果の違いが小さいことが示されている。すなわち、学生を被験者とした実験で得られた結果は、開発者に対して適用してある程度同様の結果が得られることが期待できるといえる。また、本研究で用いたタスクの作業量それほど大きくないが、実験時間の制約などにより、本研究と同程度かそれよりも小さいタスクの作業量を被験者に課している研究も多く存在する[8][25]。

ゲーミフィケーションのルールは以下の2つの目的から設定した。

- ソフトウェア開発の各作業の作業時間を短縮する。そのために時間を意識するようなルールを設定した。後述するルール β , γ が該当する。
- ゲーミフィケーションの効果を高める(参加者のモチベーションを高める)。そのためにゲーミフィケーションに継続性を持たせた(前回の結果が今回の結果に反映される)場合、より効果が高まるのかを確かめられるようなルールを設定した。後述するルール γ が該当する。

4.1 レビュータスク

レビューのタスクでは、プログラムの仕様書(誤りなし)とソースコード(誤りあり)を与え、誤りの箇所と修正方法を記述してもらった。ソースコードは被験者全員が理解しているJavaで記述されている。ゲーミフィケーションのルールが作業の品質に与える影響については、レビュー指

摘数に基づいて分析する。

レビュー対象は文献[16]で示されている以下の2つとした。

- **レビュー対象A:** 数字の合計を計算して出力するプログラム。設計書が100字、プログラムが200行程度のもの。誤りが5つ含まれる。
- **レビュー対象B:** ファイルを読み込んで目的のデータを検索するプログラム。設計書が600字、プログラムが130行程度。誤りが8つ含まれる。

被験者は最初に対象Aをレビューし、次に対象Bを続けてレビューした。

ゲーミフィケーションのルールは以下の2種類を用意した。

- **ルール α :** 誤りを指摘した数で得点が計算される。予備分析に基づき、レビュー対象Aの得点は200点、レビュー対象Bの得点は250点加算することとした。
- **ルール β :** 誤りを指摘した数とレビュー時間で得点が計算される。

指摘数に基づく点数に加えて、最初に持ち点があり、レビュー時間が5秒経過するごとに5点が減点される。レビュー対象Aの持ち点は1200点、レビュー対象Bの得点は3600点とし、前述2つの合計が表示される。すなわち、時間が経過すると点数が減少するが、指摘するとその点数が回復することになる。

レビュー時には指摘内容が正しくても正しくない場合でも得点が加算されることとし、集計時に誤った指摘については加算しないこととした。この集計方法は被験者に事前に伝えた。なお、図1に示すように、どのように修正すべきかを示す必要があるため、得点を増やすことを目的として、意味のない指摘を増やすことは困難である。このため、誤った指摘に対してペナルティは与えておらず、また、指摘の正当率についても議論しない。

ルール β を適用したレビューでは、目標値を提示してモチベーションを高めるために、平均的な得点(予備実験に基づく。対象Aでは1,000点、対象Bでは3,120点とした)を被験者に示した。実際の開発においても、レビュー効率(レビュー時間÷コード行数)とレビュー指摘率(レビュー指摘数÷コード行数)を記録、蓄積しておけば、同様の目標値を提示することは可能である。ルール α の場合では、目標値は潜在的なバグ数の手掛かりとなるため提示しなかった。被験者が実験時に見る画面のイメージ(ルール α , β をそれぞれ適用した場合)を図1に示す。

レビュー対象の違いが実験結果に影響することを避けるため、以下のように被験者を2つのグループに分けてレビュー対象とルールの組み合わせを変えた。

- **グループ1:** レビュー対象Aにルール β , レビュー対象Bにルール α を適用(被験者6人)
- **グループ2:** レビュー対象Aにルール α , レビュー対

表 2 ルールによるレビュー時間の差異

グループ	対象 A	対象 B	時間比
1	0:11	0:22	2.0
2	0:15	0:29	1.9

表 3 ルールによるレビュー指摘数の差異

グループ	対象 A	対象 B	指摘数比
1	3.2	4.0	1.3
2	2.7	3.7	1.4

象 B にルール β を適用 (被験者 7 人)

4.2 コーディングタスク

コーディングのタスクでは、与えられた仕様に基づき、プログラミングをしてもらった。用いるプログラミング言語は、被験者全員が理解している Java とし、Eclipse 上でコーディングしてもらい、仕様に書いてある実行結果と一致すればコーディング完了とした。作業の品質については、実行結果と一致していれば (テストが通れば) 一定の品質が保たれているとみなし、定量的な指標を用いなかった。実際のソフトウェア開発でも、テストが通るかどうかにについては厳密にチェックされるが、コードメトリクスなどの定量的な指標を開発中に用いて、コード修正などを命じることは一般的ではない。

与えた仕様は以下の 2 つであり、比較的容易なものを Web サイト (AIZU ONLINE JUDGE [3]) から選んだ。

- **仕様 A:** 九九を出力して終了する。
- **仕様 B:** 10 個の値を読み込み、大きい順に 3 つの値を出力する。

被験者は最初に仕様 A に基づいてコーディングし、次に仕様 B にもとづいてコーディングをした。

ゲーミフィケーションのルールは以下の 2 種類を用意した。

- **ルール γ :** 得点 (回答時間) をリアルタイムで示さない。得点はタスク終了後に順位とともに示す。
- **ルール δ :** 得点 (回答時間) をリアルタイムで示す。レビュータスクとコーディングタスクの得点を合算したものをタスク終了後に順位とともに示す。ルール適用時の画面イメージを図 2 に示す。

どちらのルールでも、レビュータスクと同様に最初に持ち点があり、コーディング時間が 5 秒経過するごとに 5 点が減点される。仕様の違いが影響することを避けるために、被験者を 2 つのグループに分け、以下のように仕様とルールの組み合わせを変えて実験を行った。

- **グループ 3:** 仕様 A にルール δ 、仕様 B にルール γ を適用 (被験者 9 人)
- **グループ 4:** 仕様 A にルール γ 、仕様 B にルール δ を適用 (被験者 5 人)

4.3 アンケート

コーディングタスク実施直後に、以下のアンケートを実

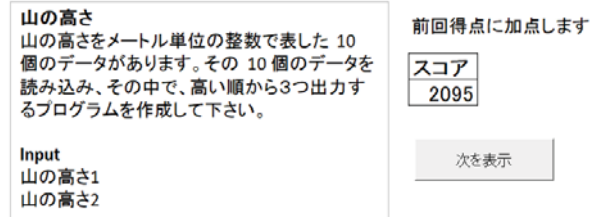


図 2 コーディングタスク時の画面イメージ (ルール δ を適用)

表 4 ルールによるコーディング時間の差異

グループ	対象 A	対象 B	時間比
3	0:09	0:34	3.7
4	0:12	0:31	2.6

施した。各項目について、主観に基づき 5 段階 (1 が最も当てはまらない, 5 が最も当てはまる) で評価してもらった。

- Q1: ゲーミフィケーション (ランキング) により、作業効率ややる気に差が出たと感じたか?
- Q2: スコアが出る場合と出ない場合で、作業効率ややる気に差が出たと感じたか?
- Q3: 前回の得点に加算する場合としない場合で、作業効率ややる気に差が出たと感じたか?

なお、アンケートではコーディングを好むか、レビューを好むかについてはアンケートしていない。タスクに対する好みによって、作業効率が異なる可能性はある。ただし以降の分析では、各個人の作業成果がルールにより相対的にどのように変化したかを評価しているため、タスクの好みによる作業効率の違いには影響を受けない。

5. ルールと作業結果との関連

5.1 コードレビューへの影響

レビュー時間: レビュー時間を表 2 に示す。太字はルール β (時間を考慮) を適用した場合を示す。もしルール β に時間短縮効果があるならば、グループ 1 ではレビュー対象 A のレビュー時間が相対的に短くなり、逆にグループ 2 ではレビュー対象 B の時間が短くなる。ここで、対象 A のレビュー時間を分母、対象 B のレビュー時間を分子としたものをレビュー時間比とする。このとき、ルール β に効果があるならばグループ 1 では比の値が大きくなり、グループ 2 では比の値が小さくなる。

ただし実際には、表に示すようにグループ 1 とグループ 2 のレビュー時間比にはほとんど差がなかった。このことから、ルール β は時間を短縮する効果がないと考えられる。

レビュー指摘数: レビュー指摘数を表 3 に示す。太字はルール β (時間を考慮) を適用した場合を示す。ここで、レビュー時間比と同様にしてレビュー指摘数比を定義する。

表 5 各質問項目の基本統計量

	Q1	Q2	Q3
平均値	3.5	3.7	3.1
標準偏差	1.3	1.0	1.3
2 以下の回答割合	29%	14%	43%

表 6 Q1 との相関係数

(a) コーディング時間		
仕様 A	仕様 B	合計
-0.04	-0.47	-0.25
(b) レビュー時間		
対象 A	対象 B	合計
-0.53	0.37	0.06
(c) レビュー指摘数		
対象 A	対象 B	合計
-0.07	0.34	0.29

このとき、もしルール β が指摘数を減らしてしまう効果があるならば、グループ 1 では比の値が大きくなり、グループ 2 では比の値が小さくなる。

表に示すように、グループ 1 とグループ 2 のレビュー指摘数比の差はほとんどなかった。すなわち、ルール β を適用しても、指摘数はほとんど増減しないといえる。

上記の実験結果より、コードレビューにおいて時間を意識させるようなゲーミフィケーションのルールを導入しても、時間短縮の効果はあまり期待できないといえる。これは、レビューではコードを一通り読む必要があり、作業を省略するようなことができず、かつ、コードを読む速度を速めることは難しいためであると考えられる。また、作業の品質（レビュー指摘数）に対してルールが特に悪影響を与えていなかった。このことから、コードレビューにおいて時間を意識させるようなルールは導入しても効果がないと考えられる。

5.2 コーディングへの影響

コーディング時間を表 4 に示す。太字はルール δ （時間表示、前回得点を考慮）を適用した場合を示す。仕様 A のコーディング時間を分母、仕様 B のコーディング時間を分子としたものをコーディング時間比とする。レビューの分析と同様に、もしルール δ に時間短縮効果があるならば、グループ 3 では仕様 A のコーディング時間が相対的に短くなり、グループ 4 では逆になる。そのため、グループ 3 では比の値が大きく、グループ 4 では比の値が小さくなる。

表においても、グループ 3 のコーディング時間比がグループ 4 のものよりも大きくなっていった。このことから、ルール δ は時間を短縮する効果があると考えられる。6 節で後述するように Q2 の評価が高かったことから、ルール δ の時間を意識させる部分が主に効果的であったと考えられる。コーディングの場合、ある実装を行うには複数の方法があるため、時間が掛からない（または思いつきやすい）

コーディング方法が選択された可能性がある。

実験結果より、作業時間を意識させるようなルールを設定することにより、コーディング時間を短縮できる可能性がある。ただし、作成されたコードの動作が正常でも、コードの内部構造（実装方法）がどの程度適切なのかが不明であり、その点についてはさらに分析を行う必要がある。

6. 個人の嗜好と作業結果との関連

6.1 ゲーミフィケーションの嗜好との関連

ゲーミフィケーションの主観的評価：被験者がゲーミフィケーション及びそのルールを主観的にどのように評価しているかを分析した。質問項目 Q1 から Q3 について、平均、標準偏差、及び回答が 2 以下となった割合を表 5 に示す。Q1, Q2 については、平均値が 3.5 を上回っており、2 以下の評価も多くないことから、比較的評価が高いといえる。Q3 については、43%が 2 以下の評価をしていることから、個人によってルールに対する評価差があるといえる。

主観的評価と作業結果との関係：質問項目 Q1, すなわちゲーミフィケーションを好むことと、作業結果に関連が見られるかどうか（ゲーミフィケーションを好むと、レビュー時間やコーディング時間などが改善するかどうか）を分析した。そのために、Q1 と各作業結果との相関係数を算出した。分析では、変数間の関連を分析する際、外れ値の影響を避けるためにスピアマンの順位相関係数を用いた。以降では単に相関係数と記述する。

結果を表 6 に示す。ここで合計とは、例えば仕様 A と仕様 B のコーディング時間の合計を指す。対象 A, 仕様 A は規模が小さく、作業時間のばらつきが大きいいため、それぞれの合計に着目する。レビュー時間は Q1 とほとんど関係がなかった。コーディング時間は弱い負の相関があり、レビュー指摘数は弱い正の相関があった。このことから、ゲーミフィケーションを好む場合、一部の作業効率などが高まる、すなわちコーディング時間が短くなり、レビュー指摘数が増える傾向が若干ながら存在する可能性がある。

前回結果の嗜好への影響：前回のゲーミフィケーションの結果、すなわち、レビューにおけるゲーミフィケーションの順位（結果）が良かった場合に、ゲーミフィケーションの嗜好が変化するかどうかを分析した。そのために、質問項目 Q1, Q3 と前回の順位との関連を確かめた。その結果、それぞれの相関係数は 0.07, 0.01 となり、非常に小さな値となっていた。このことから、前回の結果によってゲーミフィケーションそのものやゲーミフィケーションのルールに対する嗜好が変化しているわけではないといえる。

6.2 時間割引率との関連

主観的評価との関係：時間割引率と、ゲーミフィケーションの主観的評価に関連があるかどうかを分析した。そのために、質問項目 Q1~Q3 の相関係数を算出した。表 7 に結果を示す。Q1, Q2 との相関係数は小さく、Q3 との関連

表 7 時間割引率と各質問項目との相関係数

Q1	Q2	Q3
-0.00	0.14	0.26

表 8 ルールによるレビュー時間の差異 (層別)

(a) 時間割引率が低いグループ

グループ	人数	対象 A	対象 B	時間比
1	4	0:10	0:21	2.1
2	3	0:16	0:27	1.7

(b) 時間割引率が高いグループ

グループ	人数	対象 A	対象 B	時間比
1	2	0:13	0:24	1.8
2	4	0:15	0:30	2.0

表 9 ルールによるレビュー指摘数の差異 (層別)

(a) 時間割引率が低いグループ

グループ	人数	対象 A	対象 B	指摘数比
1	4	3.0	4.0	1.3
2	3	2.3	2.7	1.1

(b) 時間割引率が高いグループ

グループ	人数	対象 A	対象 B	指摘数比
1	2	3.5	4.0	1.1
2	4	3.0	4.5	1.5

表 10 ルールによるコーディング時間の差異 (層別)

(a) 時間割引率が低いグループ

グループ	人数	対象 A	対象 B	時間比
3	3	0:08	0:28	3.4
4	4	0:12	0:27	2.2

(b) 時間割引率が高いグループ

グループ	人数	対象 A	対象 B	時間比
3	4	0:09	0:37	3.9
4	3	0:10	0:34	3.3

表 11 時間割引率と作業結果の相関係数

コーディング 時間	レビュー 時間	レビュー 指摘数
0.28	0.28	0.53

も大きくなかった。すなわち、時間割引率とゲーミフィケーションを好むかどうか (Q1) は関連がないといえる。時間割引率はせっかちさと説明されることが多いが、スコアを出すルール (Q2) との関係も弱かった。前回の得点に今回の得点を加算するルール (Q3) については、時間割引率が高いと好む傾向が、若干ながら見られた。後述するように、ソフトウェア開発にゲーミフィケーションを適用する場面においては、時間割引率は、せっかちさと解釈しないほうがよい可能性がある。

ルールとの関係: 時間割引率が異なると、ゲーミフィケーションのルールの効果も異なるかどうかを分析した。具体的には、時間割引率の中央値で被験者を 2 グループに層別し、5.1 節、5.2 節と同様の分析を行った。データ数が少

ないことに注意する必要があるが、レビューのタスクに関しては、表 8、表 9 に示すように、それぞれのグループで、ルールによる大きな差異は見られなかった。なお、時間割引率が高いグループでは、ルール β を適用した場合 (太字の箇所) のほうが、レビュー時間が若干長く、かつ指摘数も若干多くなっていた。コーディングのタスクに関しては、表 10 に示すように、どちらのグループもルールの違いによる時間比の差が大きく、5.2 節と同様の結果が得られた。このことから、ルールの効果は時間割引率にあまり影響を受けていないといえる。

作業結果との関係: 時間割引率が、せっかちさを表すとみなすならば、作業結果と何らかの関連がある可能性がある。そこで、時間割引率と作業結果との相関係数を算出した。この時、レビュー時間は、対象 A と B の時間の合計とし、相関係数を求めた (対象 A の規模が小さく、作業時間のばらつきが大きいため)。その他の作業結果も同様である。

結果を表 11 に示す。前述の結果より、ゲーミフィケーションのルールの影響は小さいとみなし、ルールによる層別などはしていない。相関係数が正となっていることから、時間割引率が高い場合、比較的急がずに作業をし、その結果、レビュー指摘数も増えていることになる。時間割引率は、競争心と弱い負の相関があることが指摘されており [11]、時間割引率が高い被験者は競争心が弱く、そのために作業を急がなかった可能性がある。少なくとも、ゲーミフィケーションを適用したソフトウェア開発タスクにおいては、時間割引率は性急さを示していると考えないほうがよいといえる。

7. おわりに

本研究では、ゲーミフィケーションのルールの差異が作業時間などに影響するかどうか、また、ゲーミフィケーションの効果は、個人の嗜好の差異に影響されるのかどうかを実験により確かめた。実験の結果、以下の傾向が見られた。

- コードレビューにおいて時間を意識させるルールを導入しても、時間短縮の効果はないが、レビュー指摘数が大きく減少するような傾向も見られなかった。
- コーディングにおいて時間を意識させるようなルールを導入すると、作業時間を短縮できる可能性がある。
- 個人によりルールに対する評価が異なる場合があるため、各人の嗜好を考慮してルールを設定する必要がある。
- ゲーミフィケーションを導入したソフトウェア開発タスクにおいて、時間割引率は性急さを表すと考えるべきでない。

今後の課題は、特にコーディングにおいて、品質に対するルールの影響を分析することと、個人の嗜好とルールとの関係をより詳細に分析することである。

謝辞 本研究の一部は、日本学術振興会科学研究費補助金
(基盤 C: 課題番号 16K00113, 基盤 A: 課題番号 17H00731)
による助成を受けた。

参考文献

- [1] Akpolat, B., and Slany, W.: Enhancing software engineering student team engagement in a high-intensity extreme programming course using gamification,” In Proc. of Conference on Software Engineering Education and Training (CSEE&T), pp.149-153 (2014).
- [2] Arai, S., Sakamoto, K., Washizaki H., and Fukazawa, Y.: A Gamified Tool for Motivating Developers to Remove Warnings of Bug Pattern Tools, In Proc. of International Workshop on Empirical Software Engineering in Practice (IWESEP), pp. 37-42 (2014).
- [3] AIZU ONLINE JUDGE, <http://judge.u-aizu.ac.jp/>
- [4] Deterding, S., Dixon, D., Khaled, R., and Nacke, L.: From game design elements to gamefulness: defining “gamification,” In Proc. of International Academic MindTrek Conference: Envisioning Future Media Environments (MindTrek), pp.9-15 (2011).
- [5] Dicheva, D., Dichev C., Agre G., and Angelova G.: Gamification in Education: A Systematic Mapping Study, Educational Technology and Society, vol.18, no.3, pp.75–88 (2015).
- [6] Dubois, D., and Tamburrelli, G.: Understanding gamification mechanisms for software development, In Proc. of Joint Meeting on Foundations of Software Engineering (ESEC/FSE), pp. 659-662 (2013).
- [7] Frederick, S., Loewenstein, G., and O'Donoghue, T.: Time Discounting and Time Preference: A Critical Review, Journal of Economic Literature, vol.40, no.2, pp.351-401 (2002).
- [8] Fritz, T., Begel, A., Müller, S., Yigit-Elliott, S., and Züger, M.: Using psycho-physiological measures to assess task difficulty in software development, In Proc. of International Conference on Software Engineering (ICSE), pp.402-413 (2014).
- [9] Hamari, J., Koivisto, J., and Sarsa, H.: Does Gamification Work? - A Literature Review of Empirical Studies on Gamification, In Proc. of Hawaii International Conference on System Sciences (HICSS '14), pp.3025-3034 (2014).
- [10] Höst, M., Regnell, B., and Wohlin, C.: Using Students as Subjects—A Comparative Study of Students and Professionals in Lead-Time Impact Assessment, Empirical Software Engineering, vol.5, no.3, pp.201-214, (2000).
- [11] 池田新介, 大竹文雄, 筒井義郎: 時間割引率: 経済実験とアンケートによる分析, ISER Discussion Paper, no.638, (2005).
- [12] 晝間文彦: アンケートによる時間割引率の背景要因に関する研究, 早稲田商学, vol.432, p.1-34 (2012).
- [13] 一ノ瀬智浩, 上野秀剛: ゲーミフィケーションを構成する要素の違いと作業効率の評価, ヒューマンインタフェース学会論文誌, vol.18, no.2, pp.65-76 (2016).
- [14] Khandelwal, S., Sripada, S., and Reddy, Y.: Impact of Gamification on Code review process: An Experimental Study, In Proc. of Innovations in Software Engineering Conference (ISEC), pp.122-126 (2017).
- [15] 根本紀之: ゲーミフィケーションを用いた探索的テストの効果報告, ソフトウェア・シンポジウム 2016 in 米子, pp.47-52 (2016).
- [16] 應治沙織, 上野秀剛: コードレビュー時の読み方指示によるレビュー効率の向上, 情報処理学会研究報告 ソフトウェア工学研究会, vol.2014-SE-185, no.2, pp.1-8 (2014).
- [17] Pedreira, O., García, F., Brisaboa, N., and Piattini, M.: Gamification in software engineering – A systematic mapping, Information and Software Technology, vol.57, no.1, pp.157-168 (2015).
- [18] Salman, I., Misirli, A., and Juristo, N.: Are students representatives of professionals in software engineering experiments? In Proc. of International Conference on Software Engineering (ICSE), pp.666-676 (2015).
- [19] Singer L., and Schneider, K.: It was a bit of a race: gamification of version control, In Proc. of International Workshop on Games and Software Engineering: Realizing User Engagement with Game Engineering Techniques (GAS), pp.5-8 (2012).
- [20] Smith R., and Kilty, L.: Crowdsourcing and Gamification of Enterprise Meeting Software Quality, In Proc. of International Conference on Utility and Cloud Computing (UCC), pp.611-613 (2014).
- [21] Souza, M. Veadó, L., Moreira, R., Figueiredo, E., and Costa, H.: Games for Learning: Bridging Game-Related Education Methods to Software Engineering Knowledge Areas, In Proc. of International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET), pp.170-179 (2017).
- [22] Stanculescu, L., Bozzon, A., Sips, R., and Houben, G.: Work and Play: An Experiment in Enterprise Gamification, In Proc. of ACM Conference on Computer-Supported Cooperative Work & Social Computing (CSCW), pp.346-358 (2016).
- [23] 筒井義郎, 佐々木俊一郎, 山根承子, グレグ・マルデワ: 行動経済学入門, 東洋経済新報社 (2017).
- [24] Unkelos-Shpigel, N., and Hadar, I.: Gamifying software engineering tasks based on cognitive principles: the case of code review, In Proc. of International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE), pp.119-120 (2015).
- [25] Uesbeck, P., Stefik, A., Hanenberg, S., Pedersen, J., and Daleiden, P.: An empirical study on the impact of C++ lambdas and programmer experience, In Proc. of International Conference on Software Engineering (ICSE), pp.760-771 (2016).
- [26] Vasilescu, B.: Human aspects, gamification, and social media in collaborative software engineering. In Companion Proc. of International Conference on Software Engineering (ICSE Companion), pp.646-649 (2014).
- [27] Vasilescu, B., Filkov, V., and Serebrenik, A.: StackOverflow and GitHub: Associations between Software Development and Crowdsourced Knowledge, In Proc. of International Conference on Social Computing, pp.188-195 (2013).