

## SocSaga(ソックサーガ)で双方向性のあるコンテンツを

- 2021年度から中学校プログラミング教育に、「ネットワークを利用した**双方向性のあるコンテンツ**のプログラミングによる問題解決」が追加された。しかし、これに対応する教材はたいへん少ない。
- **双方向性のあるコンテンツ**とは、「入力に応答する、人間にとって意味のある情報の内容」である。ロボットを動かす情報はコンテンツではないし、プレゼンや動画などは双方向性のないコンテンツである。さらにコンテンツには、「デジタル化された文字、音声、静止画、動画などのデータを**複合化**」することが求められている。
- 現代、作品・製品の多くがクリエイターとエンジニアの協力によって生まれている。どちらかといえば、双方向性はエンジニアの比重が、コンテンツはクリエイターの比重が大きい。もし両者の敷居を低くすることができれば、作品の質は飛躍的に向上する。
- ネットワークを利用したプログラミングをしようとすると、**テキストエディタで編集→ローカルファイルとして保存→アップロード→ブラウザで確認**という手順が必要となり、セキュリティの問題もからみ、煩雑・困難であった。それを解決するのが [Replit\(リプリット\)](#) などの「クラウド統合開発環境」である。
- Replit の上でWebアプリを動かすことで、ビジュアルに複合コンテンツを作り、最小限のコードで双方向性を持たせるシステムを作ってみた。それが **SocSaga** である。即、**サーガ** (冒険物語) にいざなう…との願いをこめた。
- SocSaga を使えば、ローカルファイルを管理する必要がなく、環境構築の手間なしに直接Web上のファイルを編集でき、作品の中身と動作のロジックに集中することができる。 ※【注意】 **Internet Explorer 以外のモダンブラウザ** (Chrome,Firefox,Safari,…) で、横置き画面で見ること！

1/14

コンテンツ作りへ

双方向性へ

発展へ

さあ、始めよう！

ソース socsaga-2

見出し1    A↑    A    A    B    U    S    ≡    ½    :    :    ≡    ≡    ≡    ≡    ≡    ≡    ≡    ≡

## コンテンツ作りは、ビジュアルHTMLエディタSocPad(ソックパッド)で

文字サイズは、最小 小 標準 大 最大。文字色は、いろいろと変えられる。背景色も、いろいろと変えられる。太字と、下線と、取消線（文字揃え：中央ぞろえ）。


1. 段落番号を指定すると、自動的に先頭に番号が振られる。箇条書きは、番号のかわりに・(中黒)がつく。

字下げをすると、段落全体が右に移動する（これは2回字下げしたもの）。戻すこともできる。リンクを挿入でき、クリックで別タブに表示（例→ [色使いの基本ルール：使うのは3色まで!](#)）。SocPadで作ったものをホームページとして使うことができる。

表の挿入は、	セルの結合・分割可	列・行の		← jisshu/img フォルダ内の画像一覧から画像を選んで貼り付け
列数×行数を指定して		挿入・削除可		

テスト 2 番のスライド編集エリア ↑↑↑↑↑ 水色の帯までが実行エリア ↓↓↓↓↓↓

## SocPadの使い方2

- ソックスガ用SocPadは、[ソックリッカー](#)用SocPadを流用したもの。簡易ワープロ「ワードパッド」に似た操作で簡単。
- [保存] ボタンでFirebaseのデータベースに保存するかわりに、[ソース] ボタンを使い、指示に従って直接HTMLファイル内に貼り付ける。HTMLの文法を知らなくても、見たとおりにHTML編集ができる。
- `<section></section>` を追加すれば、新規スライドが追加される。
- sectionタグ内 (`<section>` と `</section>` の間の部分) にソースを貼り付けると、スライドが保存される。
- 別のスライドを編集したいときは、`~id="henshu">(番号)</span><!-- 編集するスライド番号を入れる -->` の番号部分を書き換え、スライド編集エリアの上の  ボタンで更新する。更新すると、編集集中のものは消えるので、ソースを貼り付けたか必ず確認すること。心配であれば、メモ帳などにもソースを貼り付けておくとよい。
- 以上を繰り返して、スライドを1枚1枚作ってゆく。
  
- スライドを作ったら、「編集エリア」の下の「実行エリア」でボタンを押してスライドを切り換えてみる。切り換え動作の設定は、次の『双方向性は、4色ボタンに応じたスライド切り換えで』に従う。
- すべてのスライドの内容・動作が**完成したら、スライド番号部分に 0 を入れる**。「実行エリア」だけが表示されるようになるので、URLを公開して人に見せることができる。

## 双方向性は、4色ボタンに応じたスライド切り換えで

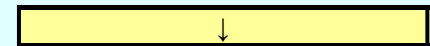
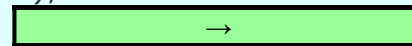
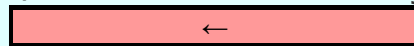
①動作は**4色のボタン**に割り当て、②飛び先は **class属性**で指定し、③必要に応じて動作を**拡張**する という [ローコードプログラミング](#) の手法で行う。「順次・分岐・反復」のスライド移動の基本命令は、xgo(数); という関数で、**数**によって動作を変える。

種類	ボタンの文字の例	既定動作	命令	sectionタグの class属性
順次	→ (次へ)	ボタン3	xgo(1);	
	← (前へ)	ボタン2	xgo(-1);	
分岐	選択肢2へ～選択肢6へ		xgo(2);～xgo(6);	class="bk2"～"bk6"
	(選択を) 抜ける		xgo(7);	class="bk7" (合流点のスライドに)
反復	↓ (次の反復ブロックへ)	ボタン4	xgo(8);	反復ブロックの開始スライドに class="hp"
	↑ (前の反復ブロックへ)	ボタン1	xgo(-8);	
	↓↓ (末尾へ)		xgo(9);	

4/14

↑↑ (先頭へ)

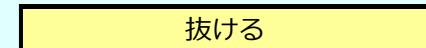
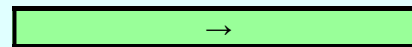
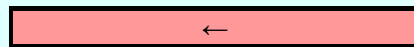
xgo(-9);



## 双方向性 2

- sectionタグには、デフォルト(規定値)で、  
`data-nav="nav0:xp+'/'+xpsum; nav1:↑;xgo(-8); nav2:←;xgo(-1); nav3:→;xgo(1); nav4:↓;xgo(8);"`  
というカスタムデータ属性が設定されている。
- `<section data-nav="nav1:先頭へ;xgo(-9);">`  
とすれば、青色のボタン1 だけ、先頭スライドへ移動するよう変更され、他のボタンはデフォルトのままとなる。nav数 のあとに コロン (:)、ボタン文字 のあとセミコロン(;)、そのあとは 命令1;命令2;命令3;…のように各命令のあとにセミコロンを付ける。二重引用符の中で引用符を使うときは、`"nav0:xp+'/'+~"` のように単一引用符を使うこと。
- 各章の始まりのスライドのsectionタグに `class="hp"` を入れておけば、デフォルトのままでも「プレゼンテーション」は十分可能。  
class属性を複数指定する場合は、`class="hp bk3"` のように半角スペースをはさむこと。
- Replitのコードエディタは、行番号の右の ▼ をクリックすることで折りたたむことができる。<section の行を折りたたんでスライドの中身を隠すと、属性の指定がやりやすい。

5/14



## 発展的な SocSaga の使い方

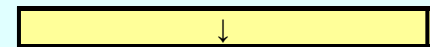
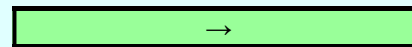
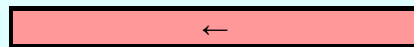
### ① スライドの背景を指定する。

- sectionタグのカスタムデータ属性に画像ファイルURLを指定。URLは、**jisshu/img フォルダ内の画像一覧**の画像を右クリックし、[画像アドレスをコピー]で。data-bg="<https://socsaga.hadano3.repl.co/jisshu/img/venndiag.svg>"

### ② x関数・x変数でボタン動作を拡張する。 ※ テストボタンの data-test = "命令文" を書き換えて、ボタン動作のテストができる。

- メッセージ表示 : xmsg('メッセージ','**入力文**'); 入力文を省略したら [alert](#)、'OK'なら [confirm](#)、それ以外は [prompt](#) と同じ動作をする。戻り値は **xm** に代入される。
- 乱数 : xran(**数**); **1~数** までのランダムな自然数を作る。**数**を省略したら **6** がはいりサイコロに使える。戻り値は、**xr** に。
- 待機 : xwait(**数**); **数**秒間待機してから次の命令を実行する。
- x変数 : **xc**【クリックしたボタン番号】、**xe**【エネルギー】、**xf**【ファイル名】、**xg**【移動後のスライドナンバー】、**xp**【現在のスライドナンバー】、**xpsum**【スライド総数】、**xs**【スコア】、**xt**【ターン数】、**xu**【ユーザーデータ】、**xuu[数]**【ユーザーデータ配列】…状態フラグとして使うのに便利な関数がある {xuuon(**数**); フラグオン、xuuoff(**数**); フラグオフ、xuutog(**数**); フラグ反転}、**xcc[数]**【ボタン無効化配列】…xuuと同様な関数でxcc[**0**]とxcc[**番**]をオンでボタン無効化、xcreset(); ですべて有効化。
- JavaScriptの命令がほぼ使えるが、コード量節約のため、[ショートハンド](#)を使うとよい。if文のかわりに→三項演算子…など。

6/14



## 発展的な SocSaga の使い方 2

③ アニメーションで画面に躍動感を与える。

- x関数の xanim(**オブジェクト**); でスライドの中の要素を時間を追って変化させることができる。アニメーションオブジェクトの書式は、[anime.js](#) に従う。戻り値は、**xa** に。

■ [Wick Editor](#) というフリーのWebアプリで手軽に Gifアニメ を作ることができる。[軽量化](#)して使うとよい。



7/14

見出しをビヨ〜ン!



絵を右へ(3秒後)左へ

## 発展的な SocSaga の使い方 3

④ 同じフォルダにあるSocSagaファイルを x関数で呼び出すことができるので、クラス全員で共同制作することができる。

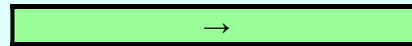
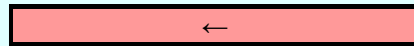
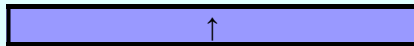
- game03.html のスライド10番で xborrow(game05); を実行すると、game05.htmlが呼び出され、xback(); で game03.html のスライド11番に戻る。xbuy(game07); を実行すると、game07.htmlに接続する（戻ってこない）。
- xborrow、xback、xbuy で他のファイルを呼び出すとき、次の x変数の値は引き継がれる。→ `xs xe xt xu xuu[]`

⑤ HTMLファイル本体をカスタマイズ（html・css・JavaScript を必要に応じて書き換え）する。文字の大きさ→ `--fontvw: 1.5vw;` methods:に新しい x関数 を追加…など。

- スライド内容の一時的な変更は、x関数の `xdisp(元の語,置換語,回数);` で可能。
- SocPad以外で作ったHTMLは、原則SocPadでの編集はできないが、sectionタグ内に貼り付けることは可能。セキュリティ上の問題がなければ、iframeタグを使って他サイトのコンテンツを埋め込むことができる。--framewidthと--frameheightで幅と高さを指定できる。
  - 無料ソフトでもCG作成ができる。Blender で3Dモデルを、Effekseer でビジュアルエフェクトを、Unity でアニメやゲームを作ることができる。Sketchfab などの CGコンテンツ共有サイトを利用することで、登録・公開ができる。次のスライドとその次のスライドに一例を示した。

※ このような発展的な使い方により、プレゼンだけでなく、クイズ・アドベンチャーゲーム・脱出ゲーム・すごろくなども作れる。

8/14



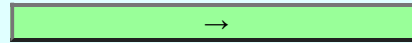


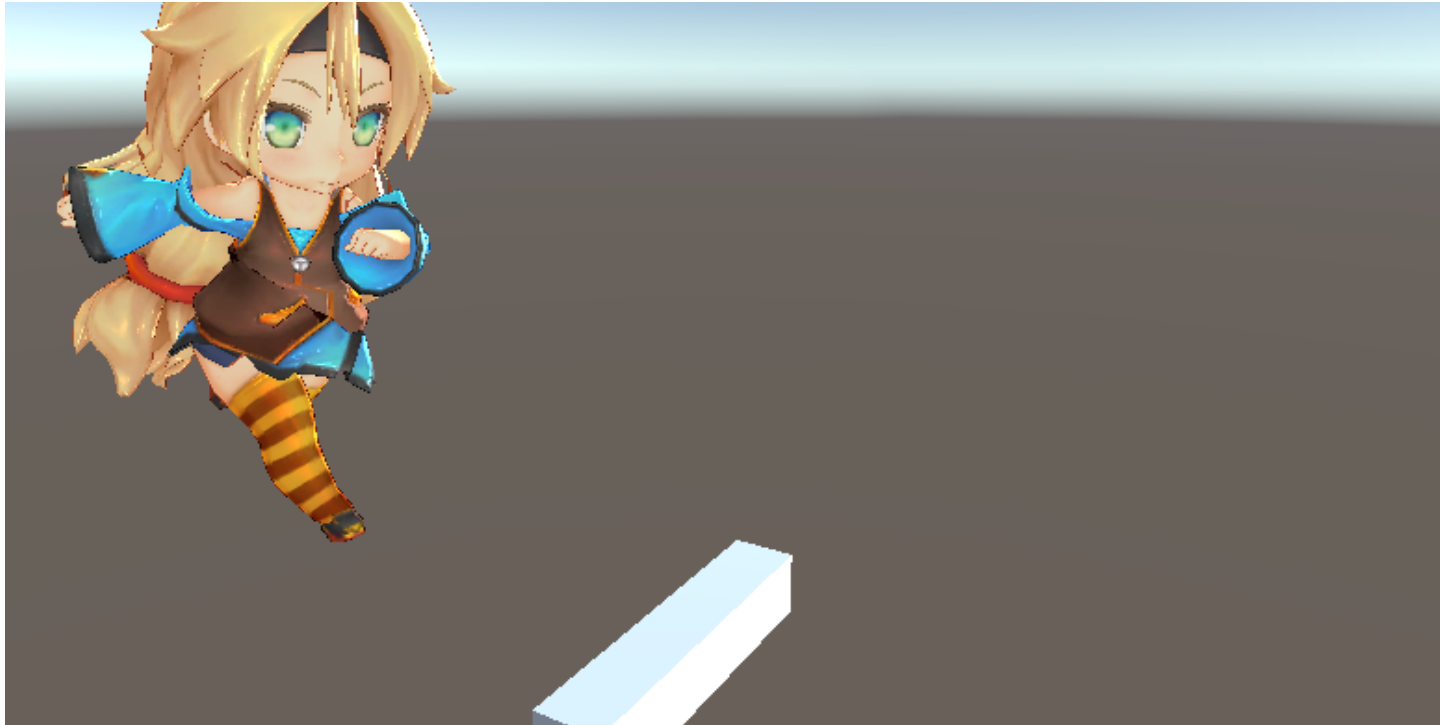


[Poly Sphere Dog Second](#) by [hadano-kouji1](#) on [Sketchfab](#)

※ 犬の3Dモデルです。マウスでグリグリできます。

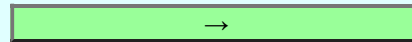
9/16





※ ユニティちゃんをタップしてハードルを飛び越えさせよう (WebAssembly版)

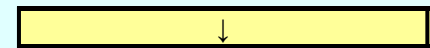
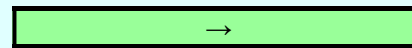
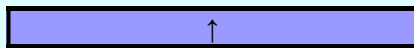
10/16



## 教材としての SocSaga について（指導者の方へ）

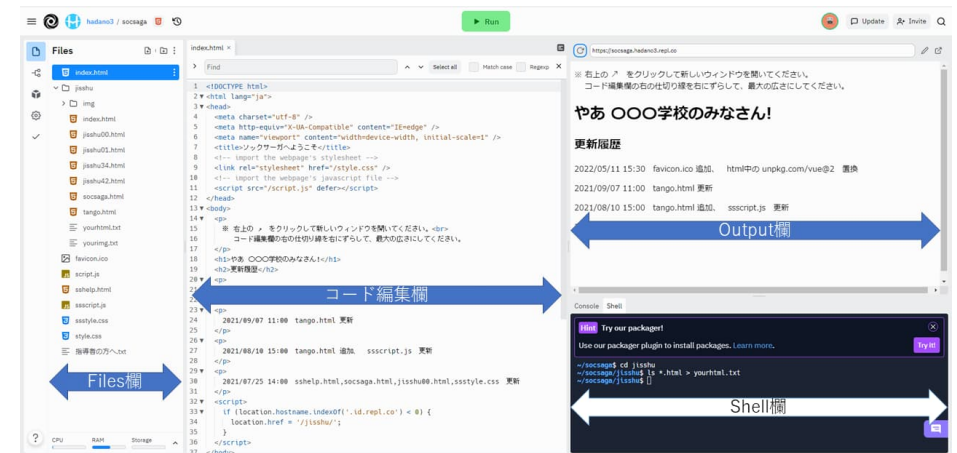
- 技術の教科書には5つの実習例が出ているが、どれも深刻な問題を抱えている。
- チャット・タイピング練習は、コンテンツが利用者まかせで貧弱であり、作り手から提供される情報がない。
- ルート検索・英単語の発音チェックは、魅力的なコンテンツではあるが、外部プログラムを呼び出して使っているだけであり、ほぼ何もプログラミングしていない。つまり、問題解決にプログラムを**使っているが作っていない**。
- 防災地図は、魅力的なコンテンツを双方向的に操作できる可能性があるという点ですばらしい。ただ、望む動作をさせようとした場合、直接 scriptタグ内を正確に書き換える必要がある。数時間の実習で理解して完成させるのは、生徒も教師も不可能に近い。
- 一方、防災地図以外の4つのアプリは、Scratch の利用を前提としている。小学生がプログラミング体験をするにはよいが、本格的な問題解決には向かない。本命は、高速・柔軟で共有・再利用できるテキストプログラミング。**SocSaga** は、Web標準技術に基づいたテキストプログラミングで主要な機能をラップすることで、入口が広く奥が深い学習を実現した。将来にわたり役立つ情報活用能力が培われる。
- 「本当にこれで授業ができるのか」という心配は無用。2017～2018年度に「デジタル作品づくり」で新指導要領を先取りして実践してみた（→3年生制作／□□中のPR）。ネットワーク環境に悩まされつつ完成させたが、その経験を活かし、最新技術をふんだんに取り入れ改良を重ねて誕生したのが **SocSaga**。5つの実習例とは違い、個性が出せるので評価しやすく、共同制作もできる。※ 画像やクリップボードの扱いが独特な **iPad** でも動かすことができる。
- 無料で、自校運用できて安心で、かゆいところに手が届くシステムは、そうそうないと自負している。**SocSaga** を使えば、「**デジタル作品づくり**」+α で取り組める。他教科や行事のまとめ・発表を兼ねたテーマにすれば、時間数不足も解消される。

11/14

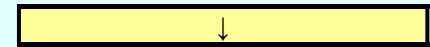
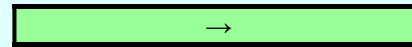
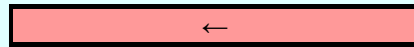
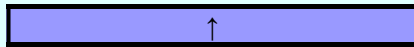


## SocSaga の始め方（指導者の方へ）


1. Googleアカウント取得。校務用とは別に授業用のアカウントがあると便利。
2. [Replit](#) に[Continue with Google]で[Sign up]し、アカウントを設定。左上の ≡ - [+Create]をクリック、Titleにプロジェクト名（socsaga1 等）、Languageは[HTML,CSS,JS]を選択し、[+Create Repl]をクリック。プロジェクトはクラス数分作るとよい。
3. JavaScriptプログラムは、基本的にhtmlファイルに組み込んで動作させる。ローカルファイルをFiles欄にドラッグ&ドロップするか、コードエディタで直接書くかする。
4. SocSagaを利用するには、[socsagaset.zipをダウンロード](#)・解凍しておく。sagaフォルダの中のものをすべてFiles欄にドラッグ&ドロップする。
5. jisshu00.html をひな形ファイルとし、ファイル名右の ≡ をクリック-Duplicate、≡ をクリック-Rename で jisshu01.html, jisshu02.html, …とクラスの生徒数分の実習htmlファイルを用意。指導者の方へ.txt の指示に従ってhtmlリストと画像リストを更新しておく。実習開始時には、右上の[ +Invite]をクリックし、メールで生徒に知らせる。  
**生徒には、次の手順で操作をさせる。**

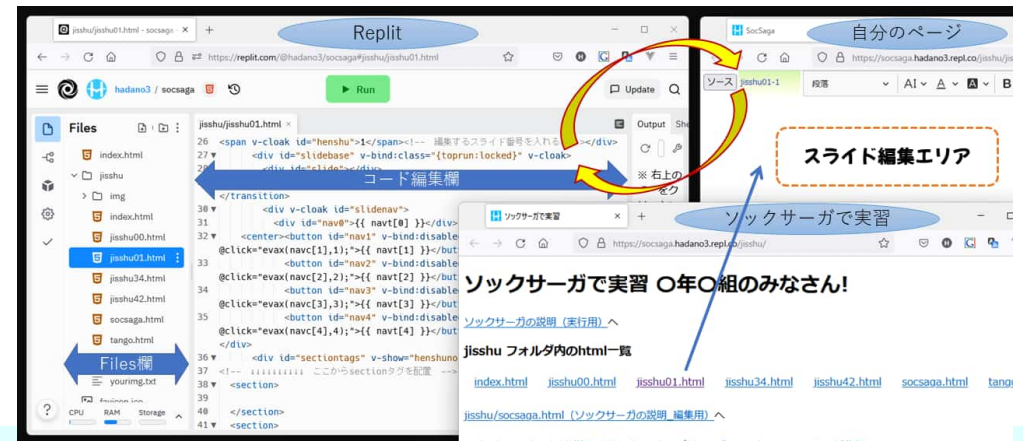


12/14

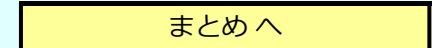
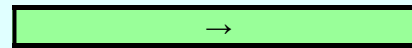
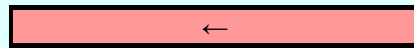


## SocSaga の始め方 (生徒へ)

6. メールが届いたら、[https://replit.com/join/\\*\\*\\*\\*\\*\\_\\*\\*\\*\\*\\*](https://replit.com/join/*****_*****) をクリックし、[Continue with Google]で[Sign up]して、プロジェクトに参加する。「Replit needs ……」に対しては、[Allow notifications]を選ぶ。
7. コード編集欄の右の Output欄の右上の ↗ (Open in a new tab) をクリックして新しいウィンドウを開く。少し待つと「ソックスアガへようこそ」から「ソックスアガで実習」に自動的に切り替わる。コード編集欄の右の仕切り線を右にずらして、広くしておくとうい。
8. [jisshu/socsaga.html](https://replit.com/jisshu/socsaga.html) (ソックスアガの説明 編集用) をクリックし、実行エリアが表示されるようにし、[コンテンツ作りへ]、[ → ]、[ → ]、…とボタンをクリックして、SocSagaの説明を読み進める。
9. SocSagaについて理解できたら、**jisshu フォルダ内のhtml一覧で [jisshu\\*\\*.html](#)** (\*\*は出席番号) をクリックして自分のページを表示する。Files欄で `jisshu/jisshu**.html` を選び、コード編集欄と自分のページとを行ったり来たりしてSocSagaの練習をする。編集するスライド番号を入力し、自分のページの  ボタンで更新。スライド編集エリアで編集後、[ソース]ボタンでコピーしたソースをコード編集欄の `section` タグ内に貼り付ける。自分のページの  ボタンで更新。正しく表示されているか、実行エリアのボタンが作動するか確認。



13/14



## まとめ : SocSagaで 双方向性のあるコンテンツのプログラミング はバッチリ !

### 双方向性

### コンテンツ

<p>利用者は、TVリモコンのように4色ボタンを押すだけでスライド切り換えができる。</p>	<p>スライドの編集と実行を同じ画面で進められる。スライド番号を 0 にすれば、実行エリアだけを公開できる。</p>	<p>HTMLの文法を知らなくても、SocPad で見たとおりにHTML編集ができる。</p>
<p>各ボタンにはデフォルト動作が設定されているので、そのままでも動く。</p>	<p>HTMLにリンクを挿入し、ホームページとして使える。他のSocSagaファイルを呼び出せ、共同制作ができる。</p>	<p>SocPadは、簡易ワープロ「ワードパッド」に似た操作で簡単。</p>
<p>ボタン動作は必要に応じて拡張できる。 Web標準に従って属性を指定するだけ。</p>	<p>工夫すれば、プレゼンだけでなく、クイズ・アドベンチャーゲーム・脱出ゲーム・すごろくなども作れる。</p>	<p>画像をReplitにアップロードして、スライドに貼り付けたり、背景画像にしたりでき、メディア複合化ができる。</p>