

SocSaga(ソックスーガ)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 [Glitch](#) などの「クラウド統合開発環境」である。
- Glitch 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！

1/16

コンテンツ作りへ

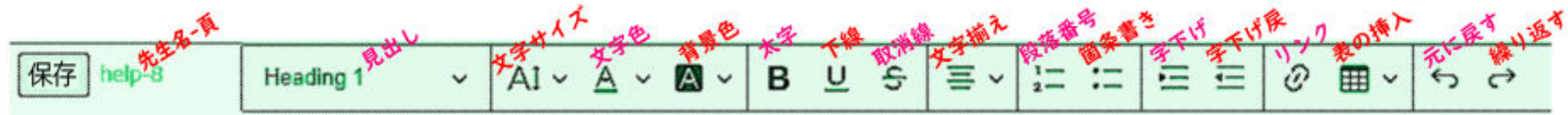
双方向性へ

発展へ

さあ、始めよう！




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→[豊後大野ジオパークへ](#)）。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		←assetsフォルダにアップロードした画像のサムネイル をクリックして現れた画像を 右クリック-画像をコピー して貼り付け
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 ↓↓↓↓↓

ソース socsaga-3 Heading 2 A↑ A A B U S ≡ ≡ ≡ ≡

SocPadの使い方 2

- ソックサーガ用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)<!-- 編集するスライド番号を入れる -->` の番号部分を書き換え、[Change URL]の右の  ボタンで更新する。更新すると、編集中的ものは消えるので、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。

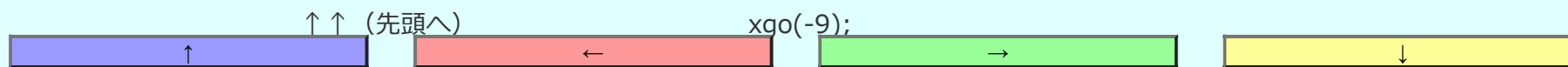
テスト 3 番のスライド編集エリア ↑↑↑↑↑ 水色の帯までが実行エリア ↓↓↓↓↓

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	<code>xgo(1);</code>	
	← (前へ)	ボタン2	<code>xgo(-1);</code>	
分岐	選択肢2へ～選択肢6へ		<code>xgo(2);~xgo(6);</code>	<code>class="bk2"~"bk6"</code>
	(選択を) 抜ける		<code>xgo(7);</code>	<code>class="bk7"</code> (合流点のスライドに)
反復	↓ (次の反復ブロックへ)	ボタン4	<code>xgo(8);</code>	反復ブロックの開始スライドに <code>class="hp"</code>
	↑ (前の反復ブロックへ)	ボタン1	<code>xgo(-8);</code>	
	↓↓ (末尾へ)		<code>xgo(9);</code>	

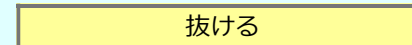
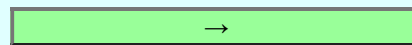
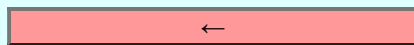
4 / 16



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。
- Glitchのコードエディタは、行番号の右の v 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。

5/16



発展的な SocSaga の使い方

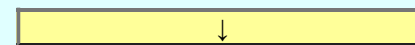
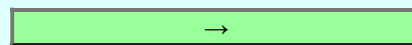
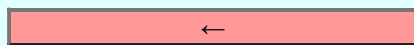
① スライドの背景を指定する。

- 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、assetsフォルダの画像をクリックし、[Copy]で。
data-bg="https://cdn.glitch.com/536402cf-1b06-4901-a66f-f5dfc1cd0ec7%2Fvenndiag.svg?v=1619573058567"

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、**xccreset()**; 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 / 16

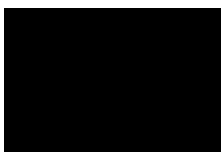


発展的な SocSaga の使い方 2

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。



7/16

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。

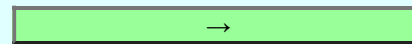
- game03.html のスライド10番で xborrow(game05); を実行すると、game05.htmlが呼び出され、xback(); で game03.html のスライド11番に戻る。xbuy(game07); を実行すると、game07.htmlに接続する（戻ってこない）。
- xborrow、xback、xbuy で他のファイルを呼び出すとき、次の x変数の値は引き継がれる。→ xs xe xt xu xuu[]

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 / 16

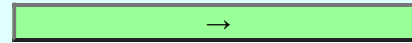


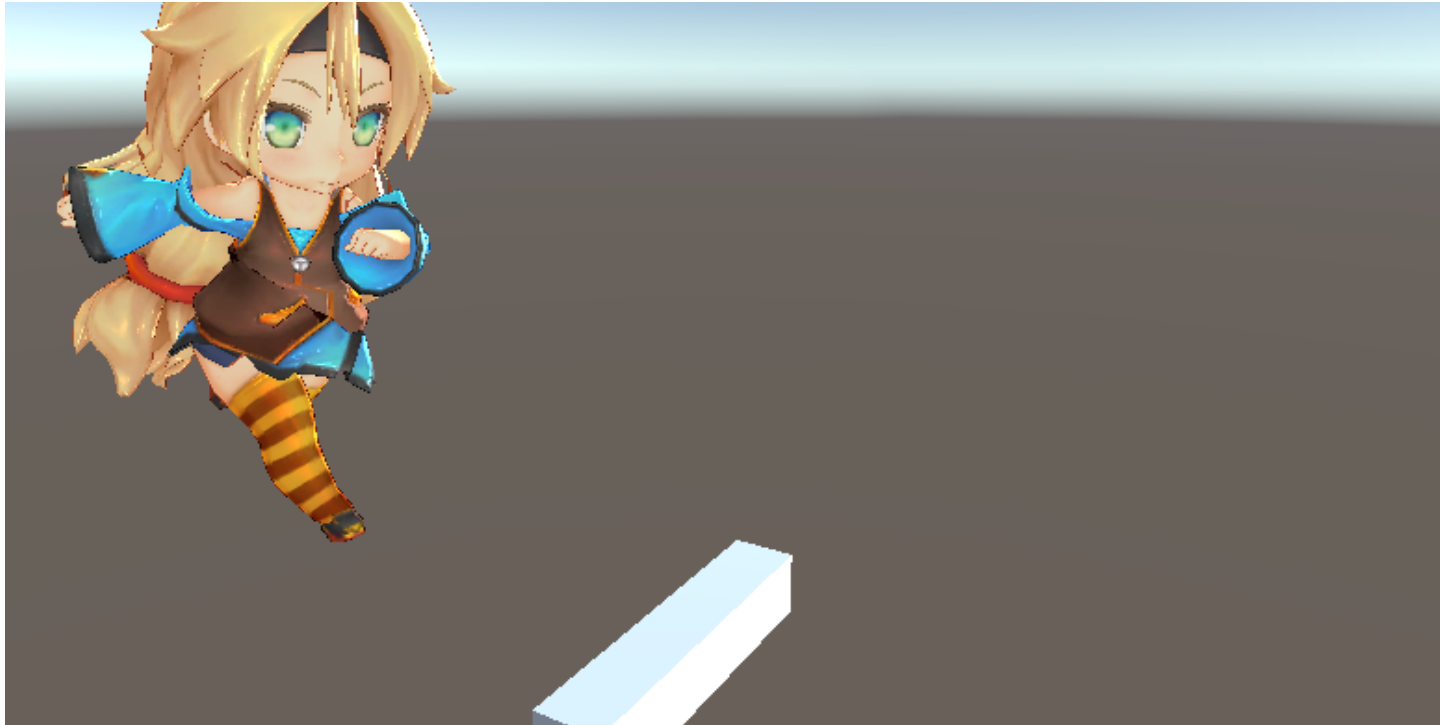


[Poly Sphere Dog Second](#) by [hadano-kouji1](#) on [Sketchfab](#)

※ 犬の3Dモデルです。マウスでグリグリできます。

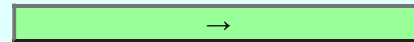
9/16





※ 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)

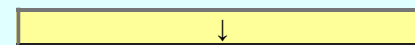
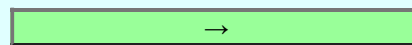
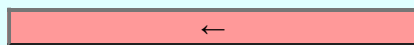
10/16



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／清川中の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。
- 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**」+**a** 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。

11/16



SocSaga の始め方 (指導者の方へ)

1. Googleアカウント取得。学校代表とは別に授業用のアカウントがあると便利。
2. [Glitch](#) に学校のアカウントで[Sign up]し、[New Project]-[hello-webpage]を選択。
3. JavaScriptプログラムは、基本的にhtmlページに組み込んで動作させる。ローカルファイルを一つずつアップロードするか、Glitchエディタで直接書くかする。SocSagaを利用する場合は、[socsagaset.zipをダウンロード](#)・解凍しておく。socsagasetフォルダの中のものを使用してゆく。saga.zip はGlitch設置用ファイルのかたまり。
4. 3では「アップロードは一つずつ」としているが、複数ファイルをアップロードする方法もある。また、画像ファイルなどは assetsフォルダに入るが、以下の手順 (①⑥以外) でカレントディレクトリに入れられる。→ [Glitchで複数ファイルアップロードできる?](#)

①アップロードしたい複数ファイルを入れたフォルダを右クリック-送る-圧縮(Zip形式)フォルダー で圧縮する(saga.zipは圧縮済み)。

②この saga.zipファイルをアップロードすると、assetsフォルダに入る。クリック-[Copy]で、URLをコピーしておく。

③[Tools]-[Terminal]で、ターミナルを開き、コマンドが打てるようにする。

④\$ wget -O saga.zip https://cdn.glitch.com/~saga.zip?v=~ (右クリック-貼り付け で先ほどのURLを貼り付ける)

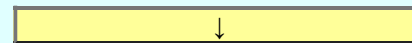
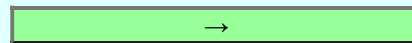
⑤これで、saga.zipファイルがカレントディレクトリ (/app) に入る。

⑥\$ unzip saga.zip (Zipファイルを解凍する)

⑦\$ refresh (変更を画面に反映させる)

⑧必要に応じて ファイルの右の : をクリック-Rename でファイル名を変えたりフォルダ名を前につけたりする。

12/16

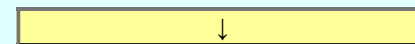
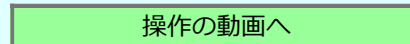
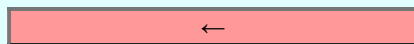


SocSaga の始め方 (生徒へ)

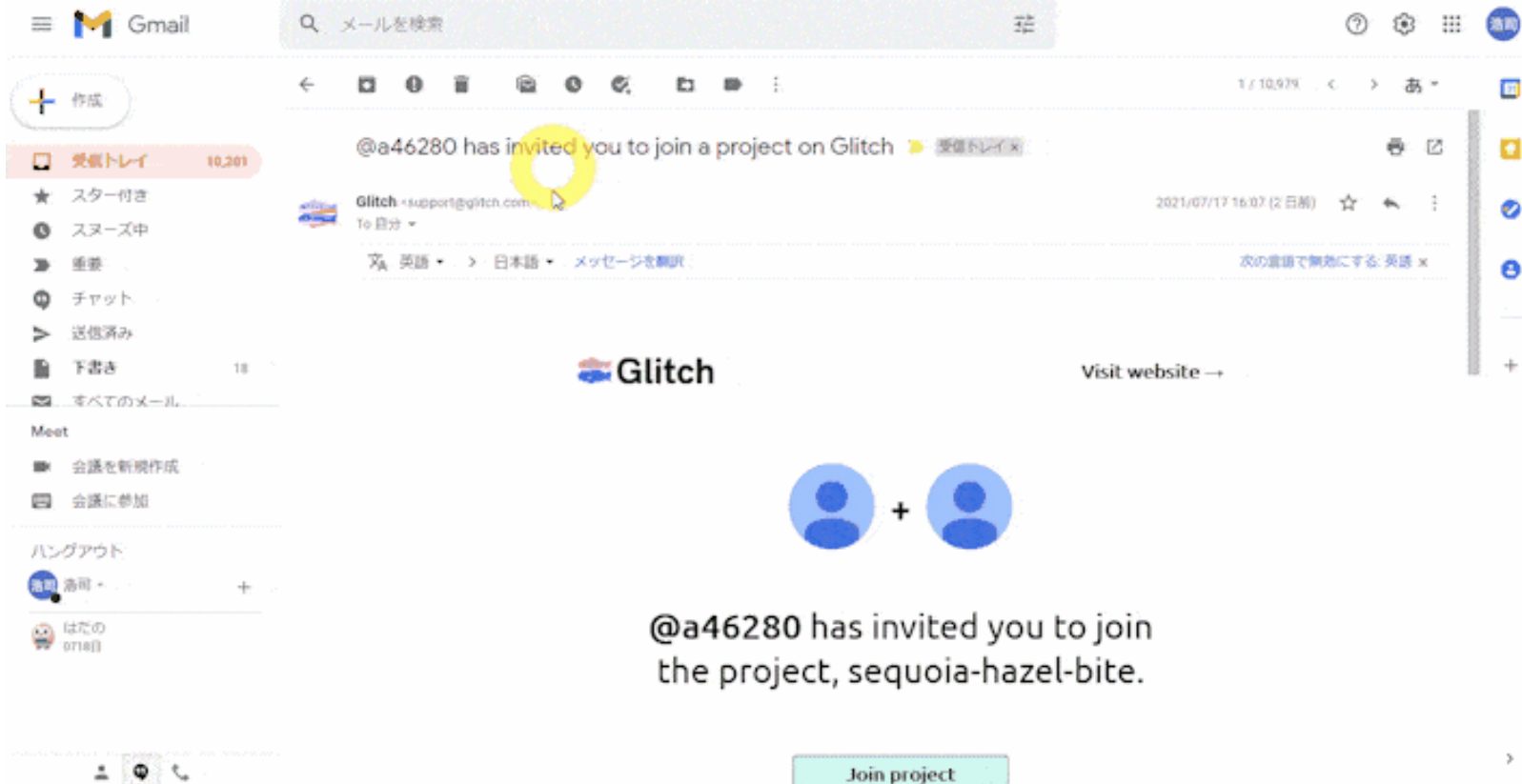
5. 先生が、jisshu00.html といったひな形ファイルを作り、Duplicate-New name で jisshu01.html, jisshu02.html, …と生徒数分用意。クラスごとにフォルダを別に。実習開始時には、🌐Shareで Invite project members to edit your code.の下の欄に授業用アカウントを入れ、[Send invite]で生徒に知らせる。生徒は、以下の手順で操作をする。 ※ 次のスライドの動画で操作イメージをつかむとよい。

- ① Glitchからメールで～join a project on Glitch～が届いたら、[Join project]をクリックし、プロジェクトに参加する。
- ② 🗨️ Show-Next to The Code で、右にプレビュー画面を表示する。左上のプロジェクト名をクリックし、Editor Settings を [☐ Refresh App on Changes]、[☑ Wrap Text]と設定しておく。
- ③ 左の列の jisshuフォルダのsocsaga.html をクリックし、右上の[Change URL]で /jisshu/socsaga.html を入力。左の列と右のプレビューは自動的には連動しないので注意。
- ④ 左のファイルリストと真ん中のコードエディタの幅を狭めて、実行エリアが大きく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の説明プレゼンを読み進める。
- ⑤ SocSagaについて理解できたら、jisshu/socsaga.html のかわりに jisshu/jisshu**.html (**は出席番号) を選び、実際にコンテンツを入力したり、ボタン動作を変更したりして、SocSagaの練習をする。いちばん上の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エディタのsectionタグ内に貼り付ける。[Change URL]の右の 🔄 ボタンで更新し、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。

13/16



テスト 1 番のスライド編集エリア ↑↑↑↑↑ 水色の帯までが実行エリア ↓↓↓↓↓↓



14/16

↑

←

→

まとめへ

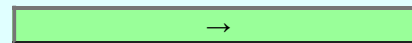
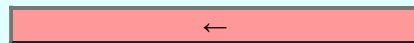
SocSaga の始め方（指導者の方へ 2）

6. 練習である程度使いこなせるようになったら、テーマに基づいて作品づくりをする。完成したら、スライド番号を 0 にして実行エリアのみとする。このまま公開することもできるが、**待ち時間短縮・不用意な改変防止**のため、次のようにする。

7. 生徒全員分が完成したら、教師が以下の手順でデプロイ〔サーバー上に配置〕して公開する。

- ①公開のためにFirebaseにログインする。Googleの Firebase は、ホスティング（Webページ公開）・ストレージ（データ保管）・データベース（データ管理）などの有用なサービス群（基本無料）。firebase のページの右上のユーザー名をクリックし、学校代表のアカウントになっているか確認。
- ②[コンソールへ移動]をクリック、Firebaseプロジェクト-[+ プロジェクトを追加]をクリック。
- ③socsaga と入力し、[続行]、☑に同意します-[プロジェクトを作成]…[続行]で、socsaga-***** などの新規プロジェクトが作られる。
- ④Glitchエディタで firebase.json を選択し、([Sign In To Firebase])-[Select Firebase Project]で socsaga を選択、[Deploy to Firebase]ボタンでデプロイ。Deploy complete! が出るまで待つ。エラー時は、Sign Out→Sign Inしてリトライ。
- ⑤[View Firebase App]ボタンでfirebase上に公開されたWebページを見ることができる。index.html以外を見るには、https://socsaga-*****.firebaseapp.com/ のあとを打ち換えるか、index.html内に相対リンクを貼るかする。

15/16



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Glitch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>

16 / 16

[先頭へ](#)

←

→

↓

Share

New File

assets

- img/
- jishu/
 - burner.html
 - burner2.html
 - burner3.html
 - geo01.html
 - index.html
 - jishu00.html
 - jishu01.html
 - jishu41.html
 - kenshu7_15.ht
 - reiji.html
 - ryuto.html
 - socsaga.html
 - speech.html
 - sticky.svg
 - svgtest.html
 - tango.html
- .env
- README.md
- fbconfig.js
- firebase.json
- index.html
- indexOld.html
- package.json
- script.js
- sshhelp.html
- sssscript.js
- ssstyle.css
- style.css

```

20 <div id="editor"></div>
21 <div id="app">
22   <span id="kakutei" v-show="!locked" v-cloak><button @click="savetext">ソース</button>&nbsp;<span
id="TeachP" v-cloak>{{ teachp }}<span v-show="asterisk">&nbsp;</span></span></span>
23   
24   <div id="footnote" v-show="!locked" v-cloak>
25     <button id="testbtn" @click="evax(testcmd);" v-cloak data-test="xmsg('音声認識の動作テストをします。[OK]
のあと、マイクに向かって何か英単語を発音してください。');xhear();">テスト</button><!-- data-test の中身を書き換
える -->
26   <span v-cloak id="henshu">4</span><!-- 編集するスライド番号を入れる --></div>
27   <div id="slidebase" v-bind:class="{toprun:locked}" v-cloak>
28     <div id="slide"></div>
29     <transition name="fade"><div id="slideveil" v-show="veil"></div></transition>
30     <div v-cloak id="slidenav">
37 <!-- ↓↓↓↓↓↓↓↓↓↓ ここからsectionタグを配置 -->
38 <section data-nav="nav0:練習したい英単語を半角スペースで区切ったリストをsplitで配列xuuに保
持;xu=1;xs=0;xcon(0);xcon(1);xcon(3); nav1:-;; nav2:始める;xuu=xmsg('練習したい英単語を指定しま
す。','library birthday work love bag').split(' ');xgo(1); nav3:-;; nav4:終了する;xgo(7);">

```

ソース tango-4

Heading 1

AI A B U S

不正解

(> <)

```

37 <!-- ↓↓↓↓↓↓↓↓↓↓ ここからsectionタグを配置 -->
38 <section data-nav="nav0:練習したい英単語を半角スペースで区切ったリストをsplitで配列xuuに保
持;xu=1;xs=0;xcon(0);xcon(1);xcon(3); nav1:-;; nav2:始める;xuu=xmsg('練習したい英単語を指定しま
す。','library birthday work love bag').split(' ');xgo(1); nav3:-;; nav4:終了する;xgo(7);">
39 <h2 style="text-align:center;"><span class="text-tiny">教科書p246 実習例5 </span><span style="background-
color:hsl(60, 75%, 60%);color:hsl(240, 75%, 60%);">英単語の発音チェックアプリ</span></h2><ul><li><a
href="https://qiita.com/hmmrjn/items/4b77a86030ed0071f548">Webページでブラウザの音声認識機能を使おう</a> より
<ul><li>Web Speech API Speech Recognition の対応ブラウザ拡大中...</li></ul></li></ul><figure class="image"></figure>
40 </section>
41 <section class="hp" data-nav="nav0:'発音問題 '+xuu+'/' +xuu.length+' '+xuu[xu-1]; nav1:-;; nav2:練習す
る;xhear(xuu[xu-1]); nav3:-;; nav4:終了する;xgo(7);">
42 <h2 style="text-align:center;">&nbsp;</h2><h2 style="text-align:center;"><span class="text-big"
style="color:rgb(76,76,230);">英単語の発音練習をしよう！</span></h2><ol><li><span class="text-big">発音問題を見
て[練習する]をクリックしたら、すぐにマイクに向かって発音する。</span></li><li><span class="text-big">認識されたら
ら[OK]し、判定結果を見る。</span></li><li><span class="text-big">認識されないときは、数秒間待ってからもう一度行
う。</span></li></ol>
43 </section>
44 <section data-nav="nav0:'発音問題 '+xuu+'/' +xuu.length+' '+xuu[xu-1];xs=xs+10;xu=xuu.length?
xcon(2):xcoff(2); nav1:-;; nav2:次の単語;xu++;xgo(-8); nav3:-;; nav4:終了する;xgo(7); nav5:xcoff(2);">
45 <h2 style="text-align:center;">&nbsp;</h2><h2 style="text-align:center;"><span class="text-huge"
style="color:hsl(240, 75%, 60%);">正解</span></h2><h2 style="text-align:center;"><span class="text-huge"
style="color:hsl(240, 75%, 60%);">不正解</span></h2><p>&nbsp;</p>
46 </section>
47 <section class="bk2" data-nav="nav0:'発音問題 '+xuu+'/' +xuu.length+' '+xuu[xu-1];xs--;xu=xuu.length?
xcon(4):xcoff(4); nav1:-;; nav2:もう一度;xgo(-8); nav3:-;; nav4:パスして次へ;xu++;xgo(-8); nav5:xcoff(4);">
48 <h2 style="text-align:center;">&nbsp;</h2><h2 style="text-align:center;"><span class="text-huge"
style="color:hsl(0, 75%, 60%);">不正解</span></h2><h2 style="text-align:center;"><span class="text-huge"

```

テスト 4 番のスライド編集エリア ↑↑↑↑↑ 水色の帯までが実行エリア ↓↓↓↓↓↓

教科書p246 実習例5 英単語の発音チェックアプリ

- Webページでブラウザの音声認識機能を使おう より
 - Web Speech API Speech Recognition の対応ブラウザ拡大中...

Speech Recognition API - UNOFF Can I use

Browser	Version	Support
IE	12-18	Partial
Edge	79-91	Full
Firefox	22-89	Full
Chrome	25-91	Full
Safari	3.1-14	Partial
Opera	27-76	Full
Safari on iOS	3.2-14.4	Full
Opera Mini	all	Partial
Android Browser	2.1-4.4.4	Partial
Opera Mobile	12-12.1	Full
Chrome for Android	92	Full
Firefox for Android	90	Full
UC Browser for Android	12.12	Full
Samsung Internet	4-13.0	Full
QQ Browser	14.0	Full
Baidu Browser	10.4	Full
KaiOS Browser	7.12	Full
KaiOS Browser	2.5	Full

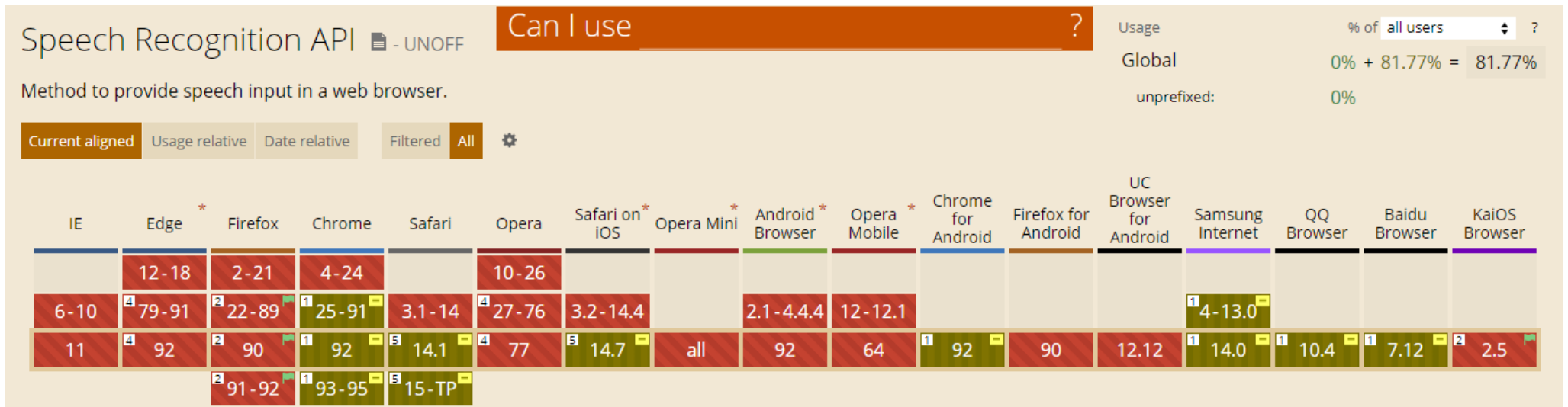
練習したい英単語を半角スペースで区切ったリストをsplitで配列this.xuuに保持

始める 終了する



教科書p246 実習例5 英単語の発音チェックアプリ

- [Webページでブラウザの音声認識機能を使おう](#) より
 - Web Speech API Speech Recognition の対応ブラウザ拡大中…



練習したい英単語を半角スペースで区切ったリストをsplitで配列this.xuuに保持

-

始める

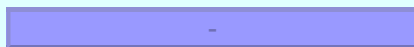
-

終了する

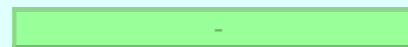
英単語の発音練習をしよう！

1. 発音問題を見て[練習する]をクリックしたら、すぐにマイクに向かって発音する。
2. 認識されたら[OK]し、判定結果を見る。
3. 認識されないときは、数秒間待ってからもう一度行う。

発音問題 1/5 library



練習する



終了する

正解

(^o^)

発音問題 2/5 birthday

-

次の単語

-

終了する

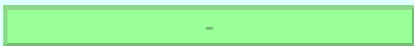
不正解

(> <)

発音問題 3/5 work



もう一度



パスして次へ



発音練習お疲れ様でした！

得点：43

