

令和6年度卒業研究報告書

学生視点による学校 PR ホームページの作成

情報技術科 佐々木 奏心

松浦 光

三上 結里

指導教員 安倍 春菜

内容

第1章 はじめに	4
第2章 研究概要	5
2.1 目的.....	5
2.2 現在のホームページ.....	5
2.3 研究の進め方.....	5
2.4 開発環境.....	11
2.4.1 Visual Studio Code.....	12
2.4.2 HTML.....	12
2.4.3 CSS.....	12
2.4.4 javascript.....	12
2.4.5 Figma.....	13
2.4.6 Google フォーム.....	13
第3章 設計	14
3.1 サイトマップ.....	14
3.2 デザイン	15
3.2.1 キービジュアル.....	16
第4章 コーディング	17
4.1 フルページスクロール.....	17
4.2 背景.....	19
4.3 アンケート・お問い合わせ.....	27
4.4 レスポンシブ対応.....	31

4.4.1 PC サイズ.....	31
4.4.2 スマートフォンサイズ.....	32
4.5 QR コード.....	34
第 5 章 内容	34
5.1 TOP.....	34
5.2 学科一覧.....	37
5.3 年間スケジュール.....	38
5.4 学生の生活.....	40
5.5 施設紹介.....	44
5.6 お知らせ.....	47
5.7 アンケート・お問い合わせ.....	48
第 6 章 考察	50
第 7 章 まとめ.....	51
第 8 章 参考文献.....	52

第1章 はじめに

現在ある本校のホームページには,学科紹介やお知らせなどで学校の紹介を行っている.

しかし,学生の学校のイベントや授業に対する意見や考えが少なく,学生視点から見た本校の情報が伝わりづらいと考えた.

そのため,現在のホームページにある情報に加え,在学生が感じるものを入学生や入学を考えている高校生やその保護者に伝えることができれば,さらに学校の良いところを理解してもらうことができる.

そして,入学する学生が増加させることができると同時に学生生活を充実するために役立つと考えた.

本研究では,現在のホームページに記載される情報を学生視点のものに改良し,新たな要素を取り入れて新入生にとって面白味のある学校 PR ホームページを作成する.

第2章 研究概要

2.1 目的

多くの学生に本校の魅力を伝えるため,学生視点で新たに学校を PR するホームページ作成する.

2.2 現在のホームページ

現在のホームページは,学科紹介や,お知らせを紹介している.しかし,それぞれ具体的な内容が紹介されておらず,授業の制作物や,学園祭の内容が伝わらないと考える.

2.3 研究の進め方

研究の進め方の図を以下に示す.

表1 研究のスケジュール

1 準備			
1.1 スケジュール決め	全員	24年 10月 3日	24年 10月 10日
1.2 寄生や他の学生にインタビュー	全員	24年 10月 3日	24年 10月 9日
1.3 情報収集	全員	24年 10月 3日	24年 10月 9日
1.4 サイトマップ作成	全員	24年 10月 3日	24年 10月 11日
2 設計			
2.1 グループ分け	全員	24年 10月 9日	24年 10月 14日
2.2 デザインを考える	全員	24年 10月 11日	24年 10月 18日
2.3 必要になるJSを調べる	全員	24年 10月 9日	24年 10月 16日
2.4 Figmaでやってみる	全員	24年 10月 9日	24年 10月 25日
2.5 文章を考える	全員	24年 10月 18日	24年 10月 28日
3 実装			

3.1 HTML 作成	全員	24年 10月 28日	24年 11月 8日
3.2 CSS を埋め込む	全員	24年 11月 8日	24年 11月 22日
3.3 JS を埋め込む	全員	24年 11月 8日	24年 11月 22日
3.4 テスト、改良	全員	24年 11月 22日	24年 11月 29日
3.5 WordPress 勉強	全員	24年 11月 29日	24年 12月 4日
3.6 WordPress 組み込み	全員	24年 12月 5日	24年 12月 13日

4 テスト・確認

4.1 テスト	オーナー名	24年 12月 13日	24年 12月 20日
4.2 見やすさなどを再確認	オーナー名	24年 12月 20日	24年 12月 23日
4.3 改良	オーナー名	24年 12月 23日	25年 1月 17日

5 再確認

5.1 卒研発表後の意見をまとめる	オーナー名	25年 2月 17日	25年 2月 19日
5.2 意見を受け、改良	オーナー名	25年 2月 19日	25年 2月 27日

① ホームページの内容決め

メンバー全員で現在のホームページを読み返した。

他に、他校のホームページを読み、また入学前に自分たちが知りたかった情報を集計した。

その後、学生視点に必要な要素を話し合い、内容を決めていった。

話し合いの一部を図 1 に示す。

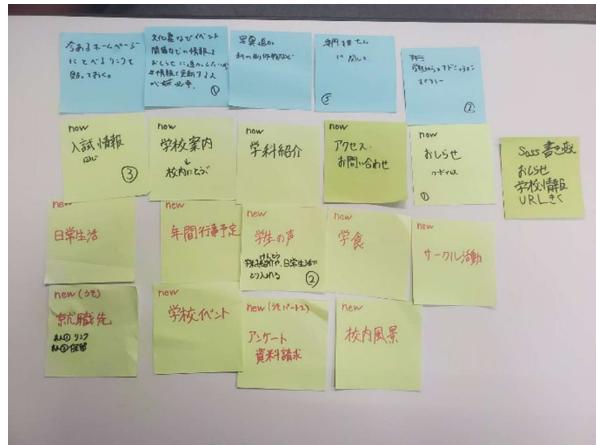


図 1 アイデア出しの様子

② インタビュー

学生視点のホームページにするには、自分達以外の多くの学生の意見を募る必要があると考えた。

インタビューは Google フォームで行った。

インタビューと回答の内容の一部を図 2 に示す。

<p>学校生活を教えてください。</p> <p>(例 8:00 電車登校 8:50 授業 12:00 友達と麻雀 13:00 授業 16:10 下校 17:00 バイト)</p> <p>2 件の回答</p> <p>8:40 学校到着 8:50 1限 12:00 昼飯、打ち合わせ 13:00 3限 16:20 学生会集会 17:10 書類記入 18:00 下校</p> <p>8:20 登校 8:30 銅の間4人東鳳戦 8:50 1限授業 10:30 2限授業 12:00 お昼休憩 & 銅の間4人東鳳戦 12:30 お昼ごはん(1年生の時は12:15) 13:00 3限授業 14:40 4限授業 16:10 授業終了 & 放課後(麻雀 か原神) 17:00 寝に帰る</p>
<p>新入生にひとこと、アドバイス</p> <p>2 件の回答</p> <p>授業と生活を両立するためには、意欲と好奇心が大切です。その二つがあれば、どちらにも全力で取り組むこ とができると思います。楽しい学生生活を送れるよう、ぜひ頑張ってください。</p>

図 2 インタビューの様子

③ 学校イベント・各科の情報収集

現在のホームページや過去の資料から情報収集を行う。

また、必要に応じて各科の学生や教員などにインタビューを行う。

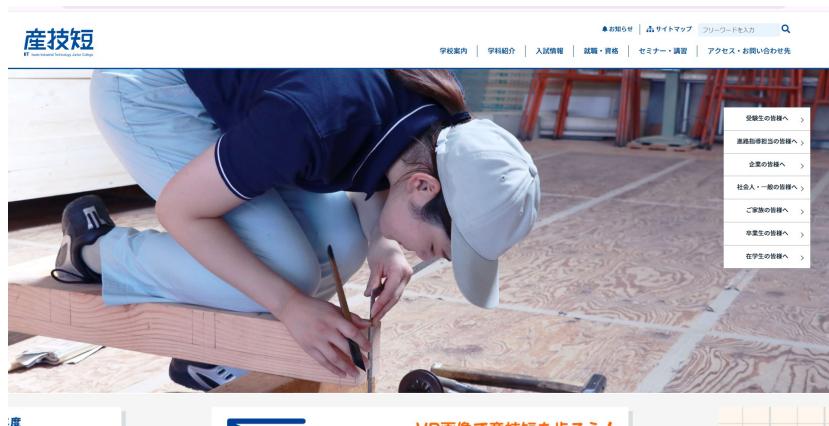
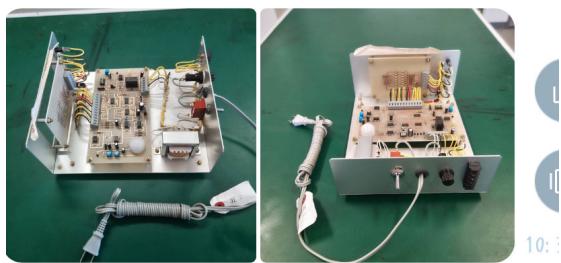


図3【岩手県立産業技術短期大学校】 <https://www.iwate-it.ac.jp/department/cis/>



電子機器組み立て2級（技能試験）の回路

1年の前期に大体の作り方を学んで、2年の前期から7月ぐらいにある技能検定本番までめちゃくちゃ作りまくって、作り方を暗記する。

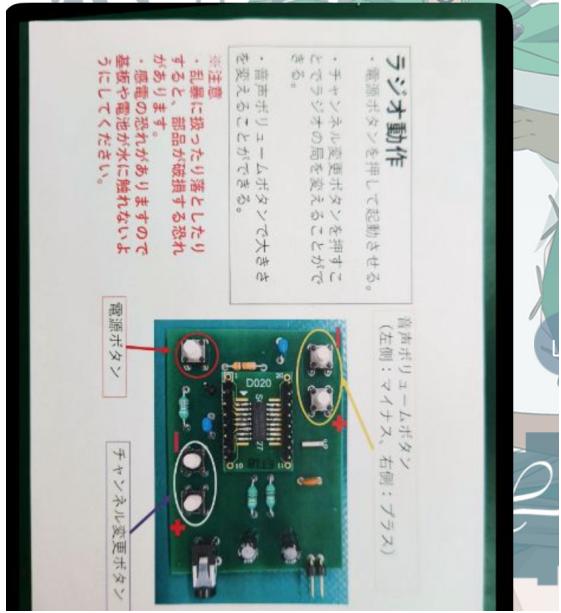


図4 電子技術科の学生からの提供

④ 演出・機能など

ホームページを見ていて面白いものになるように,様々のサイトから参考にしたい機能を全員で話し合った.

結果,以下のような意見が出た.

- ・画面遷移で特殊な動きをつける.
- ・グラフに動きをつける.
- ・背景を動画にする.

⑤ 各ページのデザイン

読者に情報がわかりやすいようにするために,文字や画像の配置,色使いなどを他のホームページを参考にし,全員が納得いくデザインにした.



図5 具体的に参考にしたページ

【岩手医科大学 WEB オープンキャンパス】

<https://www imu-admission.jp/weboc/#CAMPUSLIFE>

【北海道ハイテクノロジー専門学校】

<https://www.hht.ac.jp/>

⑥ 全体のコーディング

第一階層の大まかなデザインが完成してからデザイン担当と、コーディング担当で分担し作業を行った。また、正常に動作しているか確認しながら作業を進めた。

⑦ アンケート実装

ホームページに外部の意見を取り入れるためにアンケートを実施することにした。
アンケートを行うにあたってアンケート方法を考えていく必要があり、この当時では WordPress と Google フォームで作成する案が上がった。

⑧ 最終確認

中間発表などでいただいたご意見や、レスポンシブ対応などのデバッグ作業を行い修正していく。

2.4 開発環境

開発環境を以下の表に示す。

表2 開発環境

OS	Windows10
使用言語	JavaScript HTML,CSS
フォーム作成	Google フォーム
エディタ	Visual Studio Code
デザインツール	Figma

2.4.1 Visual Studio Code

Microsoft 社が開発した無料のオープンソースのコードエディタである。

軽量でありながら多機能で、様々なプログラミング言語の開発に対応しており、多くのユーザーに使用されている。特に HTMLCSSJavaScript を用いた Web 開発に適している。

Visual Studio Code : <https://code.visualstudio.com/>

2.4.2 HTML(HyperText Markup Language)

HTML とはウェブページを作成するためのマークアップ言語である。

ウェブページの構造を定義し、テキスト、画像、リンク、フォームなどの要素を構成するために使用される。

2.4.3 CSS(Cascading Style Sheets)

ウェブページやウェブアプリケーションのデザインやスタイルを指定するためのスタイルシート言語である。

HTML と共に使用され、CSS はコンテンツの見た目やレイアウトを指定する。

2.4.4 JavaScript

JavaScript はウェブページに複雑な機能を実装することができるプログラミング言語である。JavaScript はレイヤーケーキの 3 層目にあたり、アニメーションやスクロールする動画、ジュークボックスがある場合、ほとんど使用されている。

Javascript : <https://developer.mozilla.org/ja/docs/Web/JavaScript>

2.4.5 Figma

Figma とは、ブラウザ上でインターフェース（UI）のデザイン制作ができるデザインツールである。インターフェースのデザインを、場所を選ばずブラウザさえ起動できれば使用することができる。

<https://www.figma.com/ja-jp/>

2.4.6 Google フォーム

Google フォームとは,Google のサービスの一つとして提供されている.簡単にフォームを作成することができ,フォーム作成ツールの利用が初めての方にも使いやすいサービスになっている.

<https://workspace.google.com/intl/ja/products/forms/>

第3章 設計

3.1 サイトマップ[°]

サイトマップを以下の表に示す。

表3 サイトマップ[°]

1	2	URL
Top		/
学科紹介		/introduction
メカトロニクス技術科		/meca
電子技術科		/elc
建築科		/arc
産業デザイン科		/des
情報技術科		/it
イベント/年間行事		/event
学校スケジュール		/event
学生の生活		/voice
学生会長		/schedule1
寮生		/schedule2
一般生徒		/schedule3
施設紹介		/facility
校内風景		
学食		/gakusyoku
学生寮		/gakuseiryou
サークル		/club



3.2 デザイン

他のサイトを参考にしながら、自分達が考える本校の雰囲気を伝えることができるようなデザインを行った。



図 6 デザインの様子(Figma)

3.2.1 キービジュアル

本校の良さと考える落ち着いた雰囲気を伝えるため,全体のデザインカラーを青色にした.

しかし, 青色が与える情報の中に冷静感のほか,マイナスのイメージ,悲しみなど暗いイメージがついてしまうと考え,青の中でも明るい色を採用した.



図7 キービジュアル

第4章 コーディング

Excel・Figma で作成したワイヤーフレームを元に作成した。

構造は、フルページスクロール機能を用いた。また、PC サイズ・スマートフォンサイズにレスポンシブ対応を行った。

4.1 フルページスクロール

フルページスクロールとは、全画面を一つのページとしてスクロールすることができる機能。

ページの遷移を図 8 に示す。



↓ スクロールする



図 8 ページ遷移の様子

この実装のために,fullPage.js という jQuery のプラグインを用いた.

<div id="fullpage">要素の中にある<div>の要素において,ページ単位でスクロールを行うことができるようになっている.

フルスクリーンスクロールを利用する<div>要素を図 9 に示す.

```

44 <div id="fullpage">
45
46     <!-- begin section -->
47
48     <div class="section" data-anchor="home">
49         <div class="marquee-container">
50             <span class="marquee-text" id="text1">IT</span>
51             <span class="marquee-text" id="text2">岩手県立産業技術短期大学
52             校</span>
53         </div>
54     <!-- end section -->
55
      (省略)
220
221 </div>

```

図 9 index.html

図 10 はフルスクリーンスクロールを動かしている JS である。

navigation: true, により, 画面の横端にナビゲーションボタンを置くことで, スクロールだけでなく, ナビゲーションボタンのクリックによって見たいページまで遷移できるようにした。

navigationTooltips でページの anchor リンクとボタンの紐づけを行っている。

responsiveWidth: 300 により, 画面の横サイズが 300 以下の場合は, フルページスクロール機能が付かない且つ, ナビゲーションボタンの表示も出ないように設定した。

```

71 $(document).ready(main);
72
73     // initiate full page scroll

```

```

74      $("#fullpage").fullpage({
75          scrollBar: true,
76          responsiveWidth: 400,
77          navigation: true,
78          navigationToolips: ["home", "introduction", "event", "voice",
79          "facility", "news"],
80          anchors: ["home", "introduction", "event", "voice", "facility",
81          "news"],
82          menu: "#myMenu",
83
84          afterLoad: function (anchorLink, index) {
85              var loadedSection = $(this);
86
87              //using index
88              if (index == 1) {
89                  /* add opacity to arrow */
90                  $(".fa-chevron-down").each(function () {
91                      $(this).css("opacity", "1");
92                  });
93                  $(".header-links a").each(function () {
94                      $(this).css("color", "white");
95                  });
96                  $(".header-links").css("background-color", "transparent");
97              } else if (index != 1) {
98                  $(".header-links a").each(function () {
99                      $(this).css("color", "black");
100                 });
101                 $(".header-links").css("background-color", "white");
102             }
103
104             //using index
105             if (index == 2) {

```

```
106     /* animate skill bars */
107     $(".skillbar").each(function () {
108         $(this).find(".skillbar-bar").animate({
109             width: $(this).attr("data-percent")
110         }, 2500);
111     });
112 }
113 }
114 });
```

図 10 index.js

4.2 背景動画

ホームページの背景は,学校の風景の写真を動画にして流すことで,見る人により産
技短のイメージを持ちやすいようにした.

背景動画は,まず学校の校舎や体育館などの写真をスライドショーとして流し,そ
の他の写真をそのスライドショーの上で画面左端から右端へと流れるようにした.

流れる写真の形は丸くし,あまり大きくしないようにすることで,後ろのスライド
ショーの邪魔にならないようにした.

図 11background.js は,スライドショーの画像切り替えの JS.

図 12background-switcher.js は,スライドショーの上を流れる写真の JS である.

一度に出現する画像の数を決めずにランダムで行ってしまうと,処理が追い付かなくな
くなり,画像が 1 枚も表示されなくなってしまったため,一度に出現する画像の数を 8 枚
と設定して,動作が重くなることを避けた.

```
1 document.addEventListener('DOMContentLoaded', () => {
2   // 背景画像のリスト
3   const backgroundImages = [
4     'url("img2/photo8.jpg")',
5     'url("img2/photo4.jpg")',
6     'url("img2/photo5.jpg")',
7     'url("img2/photo6.jpg")',
8     'url("img2/photo7.jpg")'
9   ];
10
11  const intervalTime = 5000; // 画像切り替えの間隔(ミリ秒)
12  const body = document.body;
13  let currentIndex = 0; // 現在の画像のインデックス
14
```

```
15 // 背景を順番に切り替える関数
16 function changeBackground() {
17     // 背景画像を設定
18     body.style.backgroundImage = backgroundImages[currentIndex];
19     // 次のインデックスに移動(最後まで行ったら最初に戻る)
20     currentIndex = (currentIndex + 1) % backgroundImages.length;
21 }
22
23 // 初期背景の設定
24 changeBackground();
25
26 // 一定時間ごとに背景を切り替え
27 setInterval(changeBackground, intervalTime);
28 });
```

図 11 background.js

1

```

1  document.addEventListener("DOMContentLoaded", () => {
2      const items = document.querySelectorAll(".background-item");
3
4      // 画面内に表示させたい画像・動画の数を調整
5      const maxVisibleItems = Math.min(items.length, 8); // 一画面に最大 5 つ表示
6
7      items.forEach((item, index) => {
8          // ランダムな開始・終了位置(固定サイズに対応)
9          const startX = `${-100 + Math.random() * 200}vw`; // 左端から右端
10         までランダム
11         const startY = `${Math.random() * 100}vh`; // 上端から下端までランダ
12         ム
13         const endX = `${100 + Math.random() * 200}vw`; // 流れ終わる位置を
14         右方向に設定
15         const endY = `${Math.random() * 100}vh`;
16
17         item.style.setProperty("--start-x", startX);
18         item.style.setProperty("--start-y", startY);
19         item.style.setProperty("--end-x", endX);
20         item.style.setProperty("--end-y", endY);
21
22         // ランダムなアニメーション開始時間を均等に調整
23         const delay = (index % maxVisibleItems) * (18 / maxVisibleItems); // /
24         均等に遅延を設定
25         item.style.animationDelay = `${delay}s`;
26
27         // アクティブなアイテム数を制限(他のアイテムを非表示に)
28         if (index >= maxVisibleItems) {
29             item.style.display = "none";
30         }
31     });
32 });

```

図 12 background-switcher.js

また,このコードは,ホームページ作成の HTML と同じ場所で作ると,動作が重くなってしまう,ホームページを見ることが困難になってしまった.

そこで,ホームページとは別のフォルダで背景動画のプログラムを実行して,実行結果を画面録画し,それをホームページの背景動画として流す対応をした.

その動画を<body>内に入れて,フルページスクロールのどのページにおいても背景動画が流れている状態を作った.

図 13 がそのコードで,図 14 が背景動画である.

```
19 <body>
20   <header>
21     <div id="menu-button" class="menu-button">
22       <button>☰</button>
23     </div>
24   </header>
25   <!-- 目次 -->
26   <div id="menu" class="menu">
27     <ul>
28       <li><a href="#home">ホーム</a></li>
29       <li><a href="#introduction">学科一覧</a></li>
30       <li><a href="#event">学校スケジュール</a></li>
31       <li><a href="#voice">在学生の一日</a></li>
32       <li><a href="#facility">施設紹介</a></li>
33       <li><a href="#news">お知らせ</a></li>
34     </ul>
35   </div>
36
37   <div id="background-container">
38     <video class="background-item" autoplay muted loop>
39       <source src="img/haikeidouga1.mp4" type="video/mp4">
40       Your browser does not support the video tag.
```

```
41      </video>
42  </div>
```

図 1 3 index.html



図 14 背景動画

4.3 アンケート・お問い合わせ

アンケート・お問い合わせは,Google フォームを用いてデータ送信を行うようにしたことで,データベースが不要になるように作成した.

まず,Google フォームで,アンケートもしくはお問い合わせのフォームを作成し,検証ツールを使用してフォーム作成に必要な情報を抜き出し,VS コードに反映させて作成する.

作成した Google フォームのリンクを HTML に張り付けることで,アンケート・お問い合わせをすると Google フォームに回答が集まるようになっている.

図 15 は実際のフォームを反映させたコードである.

- 1 <!DOCTYPE html>
- 2 <html lang="ja">
- 3 <head>

```
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>お問い合わせ</title>
7   <link rel="stylesheet" href="query.css">
8 </head>
9 <body>
10  <header>
11    <h1>お問い合わせ</h1>
12  </header>
13  <form
14    action="https://docs.google.com/forms/u/0/d/e/1FAIpQLSf7LUiLmyrWM-
15    p6QRXL8dF-gx5ArELK160mh65E-c1Fd56axA/formResponse" method="post"
16    target="hidden_iframe">
17
18    <label for="email">メールアドレス:</label>
19    <input type="email" id="email" name="entry.986423856" required>
20
21    <label for="name">件名:</label>
22    <input type="text" id="name" name="entry.366340186" required>
23
24    <label for="message">問い合わせ内容:</label>
25    <textarea id="message" name="entry.1034729589" rows="10"
26    required></textarea>
27
28    <div class="button-container">
29      <button type="submit">送信</button>
30    </div>
31
32    <script type="text/javascript">
33      let submitted = false;
34
35      document.querySelector('form').onsubmit = function() {
36        submitted = true;
37      };
38    
```

```
34      </script>
35
36      <iframe
37          name="hidden_iframe"
38          id="hidden_iframe"
39          style="display: none"
40          onload="if(submitted){window.location='thanks.html';}"
41      ></iframe>
42      </form>
43  </body>
44  </html>
```

図 15 query.html

図 16 はアンケートとお問い合わせのリンク,図 17 は実際のページの様子である.

このアンケートとお問い合わせリンクは CSS の「position: fixed;」でページに固定し,
どのページにいても,アクセスできるようにした.

```
1007 .sidebar {  
1008     position: fixed;  
1009     right: 0;  
1010     top: 70%;  
1011     transform: translateY(-50%);  
1012     display: flex;  
1013     flex-direction: column;  
1014     gap: 10px;  
1015 }  
1016 .sidebar-button {  
1017     display: block;  
1018     width: 100px;  
1019     padding: 10px;  
1020     text-align: center;  
1021     font-size: 14px;  
1022     font-weight: bold;  
1023     color: white;  
1024     background-color: #007bff;  
1025     text-decoration: none;  
1026     border-radius: 5px;  
1027     transition: background-color 0.3s ease;  
1028 }
```

図 16 style.css



図 17 アンケート

4.4 レスポンシブ対応

4.4.1 PC サイズ

横幅 1040px 以上から PC 表示にした.

縦長の形で表示するときは, 9/16 の縦横比で背景動画を表示した.

実際のページの様子を図 18, 図 19 に示す.



図 18 PC サイズ (1040px 以上)



図 19 縦長表示 (1040px 未満)

4.4.2 スマートフォンサイズ

スマートフォンの中でも、横幅のサイズは異なり、アンケート・お問い合わせのボタンやメニュー ボタンの位置にずれが生じるため、今回は 2 つ設定した。

① 横幅 390px 以下



②横幅 430px 以下



図 20 レスポンシブ対応①

図 21 レスポンシブ対応②

```
215 <div class="sidebar">
216   <!-- アンケートボタン -->
217   <a href="survey/survey.html" class="sidebar-button">
218     アンケート
219   </a>
220   <a href="query/query.html" class="sidebar-button">
221     お問い合わせ
222   </a>
223 </div>
224 <!-- トップへ戻るボタン -->
225 <button id="scrollToTop" class="scroll-to-top">↑ TOP</button>
```

図 22 index.html

4.5 公開 URL: <https://v6.iwate-it.ac.jp/iitpr/>

QR コード



第5章 内容

5.1 TOP

ホームページの TOP は、学校名をタイトルとして表示した。(図 23 参照)

表示するタイトルは、画面右からカットインするようにアニメーションを設定した。

(図 24 参照)



図 23 TOP

```
1 window.addEventListener("DOMContentLoaded", function () {  
2     const text1 = document.getElementById("text1");  
3     const text2 = document.getElementById("text2");  
4  
5     // IIT テキストのアニメーション  
6     this.setTimeout(() => {  
7         console.log("Animation start for text1")  
8         text1.style.transform = "translateX(0)"; // 画面内に移動  
9         text1.style.opacity = 1; // 透明度を設定して表示  
10    }, 500);  
11  
12  
13     // 2 秒後に学校名を流す  
14     setTimeout(() => {
```

```
15      console.log("Animation start for text2")
16      text2.style.transform = "translateX(0)"; // 学校名を表示
17      text2.style.opacity = 1; // 透明度を設定して表示
18  }, 2500); // 2.5 秒後に学校名が流れる
19 );
```

図 24index.js

5.2 学科一覧

各科をそれぞれのイメージカラーで色分けすることで、一目でわかりやすいようにしました。これにより、閲覧者は興味のある学科を簡単に識別でき、情報の整理が促進される。(図25 参照)



図 25 学科一覧

各科の紹介ページには、授業で制作した作品を中心に掲載し、学生の実際の学びの様子を示すことで、学科の魅力を引き立てている。また、フッターには、科の選択欄を追加し、科名をそれぞれのイメージカラーにすることで、見やすく利便性を向上させた。これにより、

必要な情報に迅速にアクセスできるようになっている。(図 26 参照)



図 26 学科一覧 詳細ページ

5.3 年間スケジュール

学校行事を年間スケジュールで紹介している。

学校スケジュールの背景には主なイベントやサークルの写真を使用し、視覚的な魅力を高める。これにより、楽しみながら情報を得ることができるようになっている。(図 27 参照)



図 27 学校スケジュール

詳細ページに遷移するボタンは、マウスオーバー時に青くなるようにデザインされており、視認性が高まっている。(図 28 参照)



図 28 学校スケジュール マウスオーバー時

年間スケジュールを用いることで、学校のおおまかな行事の流れを確認できるようになり、誰にでもわかりやすくなっている。(図 29 参照)

月	イベント
4月	入学式
5月	球技大会
6月~8月	オープンキャンパス 夏季休業 インターナシップ（1年生）
9月	期末試験 楽園祭
10月	就職ガイダンス 産業技術研修（1年生）
11月	卒業研究中間発表
12月	冬季休業
1月~2月	期末試験 卒業研究発表 産技祭
3月	春季休業

図 29 学校スケジュール 詳細ページ

5.4 学生の生活

在学生の日常生活を例に、高校生活から変わる点を考慮した紹介を行っている。学生会長、寮生、一般学生の3パターンを紹介することで、学生生活の参考になるようなページに仕上げている。(図30参照)

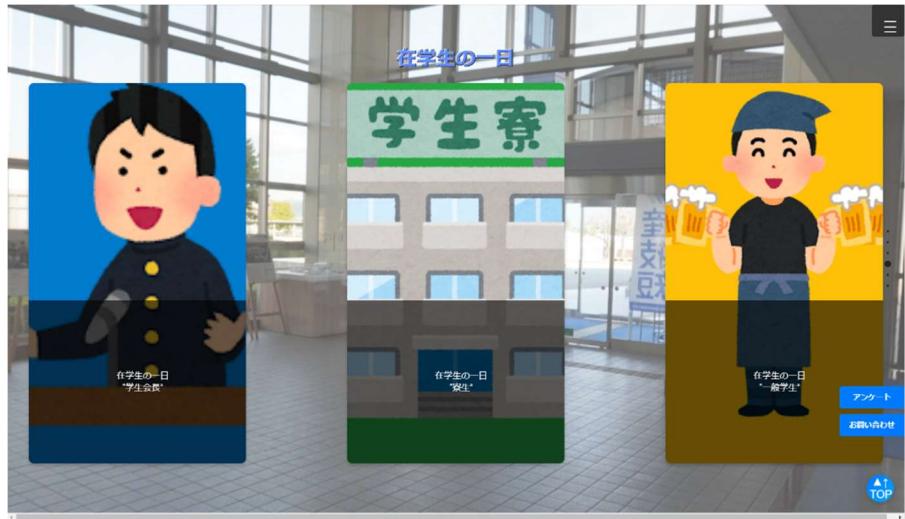


図30 在学生の一日

学生のスケジュールをわかりやすさを追求した結果、一日の流れを円グラフで表現した。(図31参照)

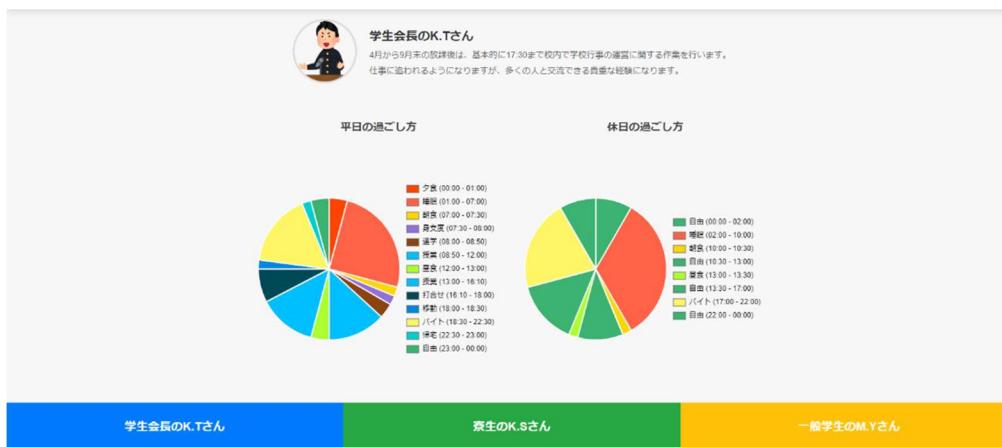


図31 在学生の一日 詳細ページ

JavaScript のプログラムを用いることで、与えられた時間を「時間」と「分」の形式にフォーマットした。(図32参照)

```

44 // 時間を「〇時間●分」の形式に変換
45 function formatDuration(duration) {
46     const totalMinutes = Math.round(duration * 60);
47     const hours = Math.floor(totalMinutes / 60);
48     const minutes = totalMinutes % 60;
49     return `${hours > 0 ? `${hours}時間` : ""}${minutes > 0 ? `${minutes}分` : ""}`;
50 }

```

図 32 schedule.js

スケジュールデータをもとに円グラフを描画し、タスクの所要時間を視覚化。グラフの描画時に回転アニメーションが適用され、視覚的なインパクトを与える。(図 33 の 67 行目から 71 行目参照)

各タスクのアクティビティと時間を含む判例を生成 (図 33 の 77 行目から 86 行目参照) し、

マウスオーバー時にタスクの詳細情報を表示 (図 33 の 92 行目から 99 行目参照) することにより理解しやすくなった。

createChart(ctx, scheduleData) 関数と window.onload イベントハンドラーのよって円グラフを表示している。(図 33 の 54 行目及び図 34 参照)

```

52 // グラフ描画関数
53 function createChart(ctx, scheduleData) {
54     const durations = scheduleData.map(item => item.duration);
55     const totalDuration = durations.reduce((sum, duration) => sum + duration,
56     0);
57     new Chart(ctx, {
58         type: 'pie',

```

```

59         data: {
60             // ラベルは削除
61             datasets: [
62                 data: durations.map(duration => (duration / totalDuration) *
63                     100), // 割合を計算
64                     backgroundColor: scheduleData.map(item =>
65                         activityColors[item.activity] || '#808080'),
66                     ],
67             },
68             options: {
69                 animation: {
70                     animateRotate: true,
71                     duration: 2000,
72                 },
73                 plugins: {
74                     legend: {
75                         display: true, // 凡例を表示
76                         position: 'right', // 凡例の位置 (例: 'top', 'left', 'bottom',
77                         'right')
78                     labels: {
79                         generateLabels: function (chart) {
80                             const data = chart.data;
81                             return data.datasets[0].data.map((value, index)
82                             => {
83                                 const { time, activity } =
84                                     scheduleData[index];
85                                 return {
86                                     text: `${activity} (${time})`, // 時間
87                                     // 範囲を含む形式
88                                     fillStyle:
89                                     data.datasets[0].backgroundColor[index],
90                                     };
91                                 });
92                             },
93                         });
94                     },
95                 },
96             },
97         },
98     ],
99 
```

```
86         boxWidth: 20, // 色のボックスの幅
87         padding: 10, // テキストとボックスの間の余白
88         color: '#000', // テキストの色
89     },
90 },
91 tooltip: {
92     callbacks: {
93         label: function (tooltipItem) {
94             const { time, activity, duration } =
95             scheduleData[tooltipItem.dataIndex];
96             return `${activity} (${time},
97 ${formatDuration(duration)})`;
98         },
99     },
100 },
101 });
102 }
```

図 33schedule.js

```

104 // グラフ描画
105 window.onload = () => {
106     const ctxWeekday =
107         document.getElementById('clockChartWeekday').getContext('2d');
108     const ctxWeekend =
109         document.getElementById('clockChartWeekend').getContext('2d');
110     createChart(ctxWeekday, scheduleDataWeekday);
111     createChart(ctxWeekend, scheduleDataWeekend);
112 };

```

図 34schedule.js

5.5 施設紹介

サークル、学食、校内案内、学生寮を紹介している。(図 35 参照)



図 35 施設紹介

学生寮では、部屋の間取りや備え付けの家具、設備、寮生活のルール、料金、セキュリティを詳しく紹介し、入寮を検討している入学生に参考になるページにしている。

学食では、学食のメニュー、写真、値段を紹介し、学生の食生活をサポートする。

どちらのページも白と青をベースに使用し、見やすさと情報のわかりやすさを重視したデザインで作成した。(図 35 及び図 36 参照)

図 35 寮

図 36 学食

サークル活動に関しては、説明文のほかに画像と動画を使用することで、活動内容をわかりやすくした。サークル長の一言を掲載し、学生の声を反映させることで、高校生に親近感を与える。(図 37 参照)

図 37 サークル

校内案内では、校内マップを各科や、本棟ごとにまとめた。また、簡易的な説明を加えてこのページで校内の理解を深められる内容にしている。(図 38 参照)

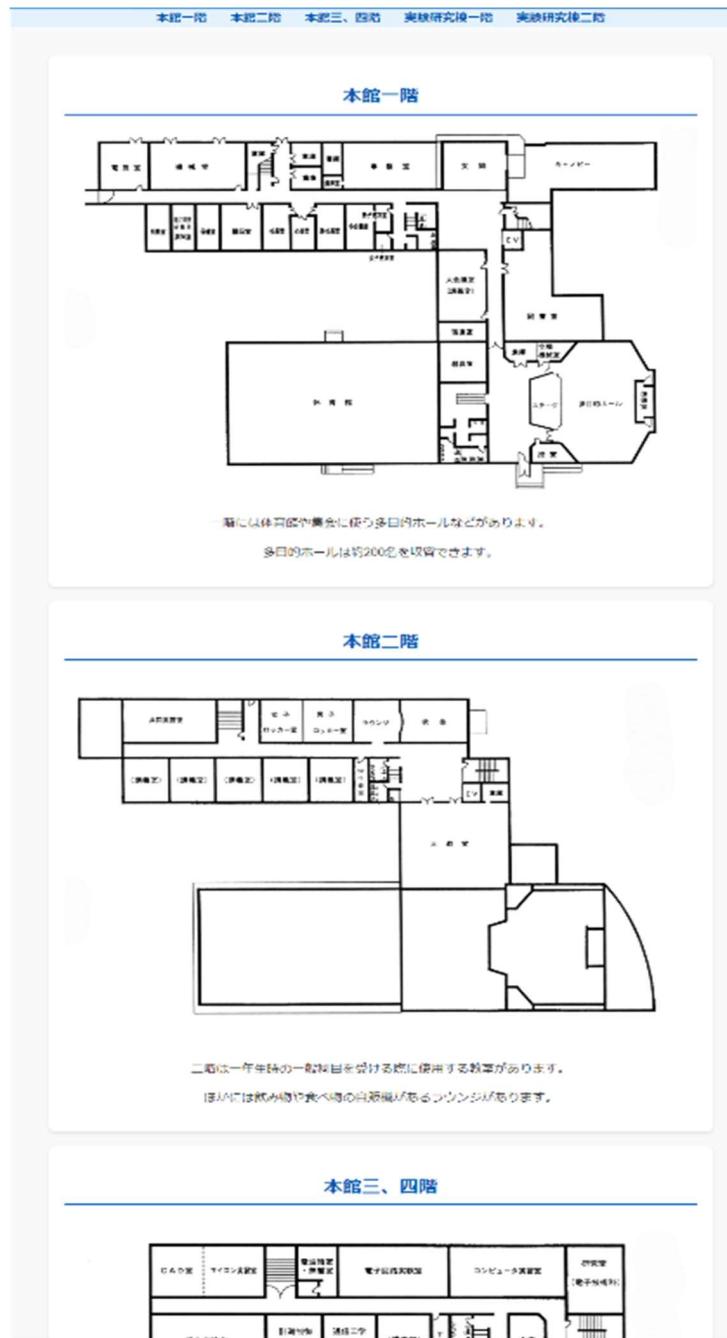


図 38 校内案内

各ページの写真のある部分はスライドショー形式で表示している。

以下のプログラムで、HTML 内の画像をスライドショー形式で表示し、8秒ごとに自動的にスライドを切り替える。
ページを閲覧している人に視覚的なインパクトを与える。

5.6 お知らせ

学校の SNS や球技大会、学園祭を年間スケジュールのページよりも詳しく紹介している。（図 39 参照）

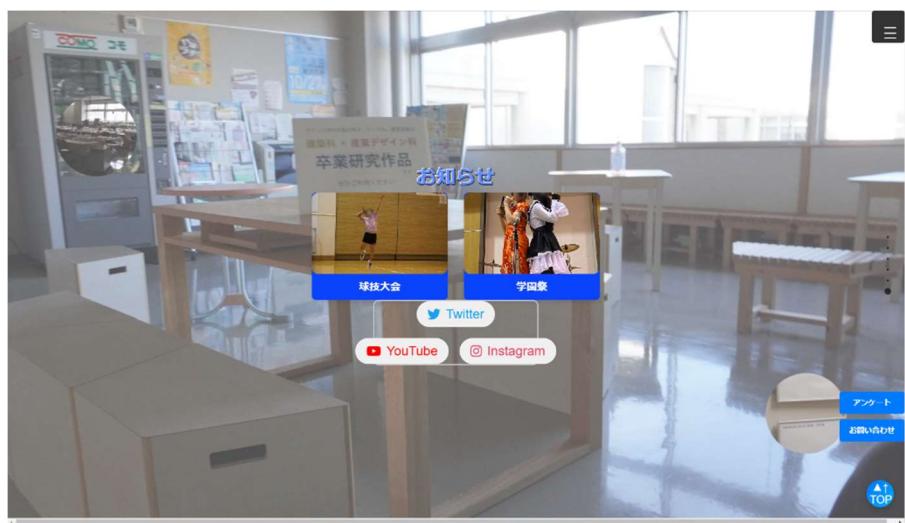


図 39 お知らせ

球技大会では例年実施している種目を、学園祭では学生だけで行われるイベント（体育館でのコンテストなど）や一般の方向けに実施されるイベント（ものづくり体験、作品の展示など）や模擬店などを紹介し、参加を検討している人の理解を促進している。これにより、学校の活動に対する興味を喚起し、参加者を増やすことを目指す。（図 40 及び図 41 参照）



図 40 球技大会



図 41 学園祭

5.7 お問い合わせ・アンケート

お問い合わせでは学校に対する意見や疑問を募集し、学生や保護者の声を反映させることを目指している。(図 42 参照)

お問い合わせ

メールアドレス:

件名:

問い合わせ内容:

送信

図 42 お問い合わせ

アンケートでは、学園祭などのイベントに対する外部の意見を集計し、今後の運営の改善に役立てるための貴重なデータを取集している。(図 43 参照)

楽園祭アンケート

楽園祭の学園祭（楽園祭）に参加したことありますか？*

はい
 いいえ

参考した理由を教えてください*

はいと答えた方のみ
 楽園祭に初心がいる
 楽園祭に興味がある
 出店が気になつた
 ステージ発表が気になつた
 イベントが気になつた
 地域の人々との交流
 楽園祭に興味がある

楽園祭の情報をどこで知りましたか？*

はいと答えた方のみ
 SNS
 友人
 楽園祭のホームページ
 ポスター

楽園祭の印象はどうでしたか？*

はいと答えた方のみ
 とてもよかったです
 よかったです
 良過ぎ
 よくなかったです
 とてもよくなかったです

特に印象に残ったイベント・出店等はなんですか？*

はいと答えた方のみ

楽園祭でぜひやってほしいイベントを教えてください*

楽園祭でぜひやってほしい出店を教えてください*

楽園祭のどの点を改善してほしいですか？*

送信

図 43 アンケート

当初はワードプレスでお問い合わせとアンケートを実施しようと考えていたが、サーバーにメールを送信することができるがサーバーからメールを受け取ることができなかったため、お問い合わせとアンケートを GoogleForm での実施に変更した。

お問い合わせ、アンケートのそれぞれのリンクをクリックすると各 GoogleForm に連携させてある HTML で作成したページに遷移し、回答を送信できるようにした。これにより、

より多くの意見を集め、学校運営に生かすことができると期待している。

第6章 考察

今回ホームページの作成に当たり,産技短の公式ホームページにはない要素を取り入れ,高校生にとって面白みのあるホームページを作成するという目標をもって励んだ。実際,普段の学生の生活の仕方や,授業の成果物など,公式ホームページにはない情報を取り入れることができたと考えている。

しかし,もともと利用しようとしていたが断念したWordPressによるアンケートの実装など,もっと勉強していればホームページに組み込むことができた内容もいくつかある。

また,現在の公式ホームページに詳しくは説明されていない学費免除の対象になる条件など,ページの中に書きたいと考えていた内容を取り入れることができずに終わってしまった。

科ごとの情報の偏りができる限りなくしたいと考えていたが,やはり,自分たちの所属している情報技術科についての情報が一番詳しくなってしまい,偏りを生んでしまった。

中間発表時に意見をいただいた,産技短を知らない人たちに対して,産技短を知ってもらうための対策が全くできずに終わってしまった。

今後は,新入生に産技短を知ってもらうための対策をする必要があると考える。

第7章　まとめ

中間発表時の課題であった,作業の分担に偏りが生まれていたことに関しては,中間発表後から改善することができ,作業の進む速さが向上した.

最初にスケジュールを作つてから,できる限りその通りに作業を進めていこうとしていたが,思つていたよりも予定通りに作業が進まなかつた.

また,それぞれの作業の進捗など話しながら作業を進めることができていなかつたことも作業分担への偏りにつながつたと考えられるため,進捗を定期的に話す必要があつたと思う.

HTML,CSS,Javascript において,まだちゃんと理解できていない部分が目立ち,先生を頼る場面が何度かあつたため,もっと自分たちで勉強していく必要があつたと考える.

第8章 参考文献

筒井美希『なるほどデザイン』(2015.8,エムディエヌコーポレーション)

佐々木恵 北川貴清 小浜愛香 窪木博士『現場でかならず使われている CSS デザインのメソッド』(2013.10,エムディエヌコーポレーション)

Mana『1冊ですべて身につく HTML&CSS と Web デザイン』(2019.3,SBクリエイティブ)

岩手医科大学 WEB OPEN CAMPUS <https://www imu-admission.jp/weboc/>

OKU LOG フルスクリーンのコンテンツをスクロール谷で切り替えるエフェクト

<https://oku-log.com/blog/one-page-scroll/>

旧質美小学校校内案内図

<https://www.taki-hj.ac.jp/guide/schoolmap.html>

<https://shitsumi.org/shogakko/52-2/>

学校・教育

<https://1guu.jp/category/industry/school/>

web design

<https://webdesignclip.com/tag/main-visual/>

web デザインポータル

<https://orient-sailing.com/design-base/category/75>

学科・コース紹介

<https://www.kobedenshi.ac.jp/course/>

神戸電子専門学校

<https://www.kobedenshi.ac.jp/>

サイトマップの作り方

<https://chot.design/>

<https://chot.design/web-director/e2277491ddc2/>

一般的なホームページ制作の流れとは？

<https://www.creal.co.jp/column/website-design/4023/>

大学サイトの UX 設計 横浜市立大学 Web リニューアル

https://loftwork.com/jp/event/20171219_visitloftwork_report