

公立はこだて未来大学 2016 年度 システム情報科学実習 グループ報告書

Future University-Hakodate 2016 System Information Science Practice
Group Report

プロジェクト名

使ってもらって学ぶフィールド指向システムデザイン

Project Name

Field Oriented System Design Learning by Users' Feedback

グループ名

保育園グループ

Group Name

Nursery School Group

プロジェクト番号/Project No.

03-B

プロジェクトリーダー/Project Leader

1014237 伊藤泰斗 Taito Ito

グループリーダー/Group Leader

1014195 大谷笙互 Shogo Otani

グループメンバ/Group Member

1014008 小嶋凌弥 Ryoya Kojima

1014114 佐藤恭祐 Kyosuke Sato

1014154 黒瀬丈哉 Takeya Kurose

1014195 大谷笙互 Shogo Ootani

1014200 神秀樹 Hideki Jin

1014212 渡邊隆広 Takahiro Watanabe

指導教員

伊藤恵 南部美砂子 奥野拓 木塚あゆみ 原田泰

Advisor

Kei Ito Misako Nambu Taku Okuno Ayumi Kizuka Yasushi Harada

提出日

2017 年 1 月 18 日

Date of Submission

Jan 18, 2017

概要

本プロジェクトでは、フィールドを実際に調査してそこから問題点を見つける。そこで見つかった問題点をICTを活用して解決する。それにより地域・社会に貢献することを目標として活動を行っている。開発手法はアジャイル開発手法を用いる。早くアプリを開発し、それに対するレビューを受けて問題解決の質をより高いものにしていく。私たちは、函館市にあるつくしの子保育園をフィールドとして設定し、つくしの子保育園での事務作業を支援するシステムを開発することを目標にして活動を行った。5月の中旬にフィールドワークを行い、そのフィールドワークからつくしの子保育園が勤務シフト表の作成や、卒園児のいる家庭に向けてのはがき作成に手間がかかっているということが分かった。これら2つの問題を解決するために、我々は勤務シフト表の作成を支援する「お助け！つくシフト」の開発と（以降、つくシフトと呼称する）、はがき作成を簡略化する「つくポス」の開発を行った。先方からの要望が強かったつくシフトの開発は前期から取り組み、これまで計3回の納品を行った。今後、4回目の納品を予定している。つくポスについては、後期から取り組み、1度の納品を予定している。つくシフトは、保育士と園児の情報を登録し、月初めのシフトを作成する。次に、自動作成ボタンをクリックすることによって、以降の日にちのシフト表が作成されるというものである。つくポスは、つくシフトで登録した園児の情報を参考に、兄弟を自動で連名にしたり、はがきを送る園児としない園児の判別を簡単に行えるようなシステムである。

（文責：大谷笙互）

Abstract

In this project, at first, we investigate on the field and find problems from field survey. We solve the problems found from field survey by ICT. Then we have action with the goal of contributing to an area. We use Agile development process which is a software development technique. We do swift app development, and develop higher quality app by being reviewed for it. We decided to work for Tsukushinoko nursery school in Hakodate. We did action with the goal of the developing the system to support office work. The Middle of May, we went there, and found that it takes much time and effort to create work shifts and make postcards for family who have graduated child. Then, we discussed the solution. To solve this two problems, we developed “Otasuke! Tsuku-Shift” (Tsuku-shift) to support to create work shifts and “Tsuku pos” to support to make postcards. Because the other party has strong demands, we developed “Tsuku shift” from the first half year, and delivered it three times until now. After this, we are scheduled to be delivered it four times. We developed “Tsuku pos” from the second half year, and delivered it one time. “Tsuku-shift” is that nursery teacher registers the information of nursery teachers and children. Then, she makes the shift at the first day. Next, the work shift is created by clicking the “auto create button”. “Tsuku pos” refers to information of nursery children registered with “Tsuku-shift”. “Tsuku pos” changes the brothers to joint signature, easily chooses whether to send postcards.

(文責: 大谷笙互)

目次

第 1 章	はじめに	1
1.1	つくしの子保育園について	1
1.2	背景	1
1.2.1	保育業界の現状	1
1.3	つくしの子保育園における現状	2
第 2 章	到達目標	3
2.1	活動目標	3
2.2	開発目標	3
第 3 章	活動プロセス	4
3.1	年間スケジュール	4
第 4 章	開発への準備	6
4.1	開発に使用したツール	6
4.2	技術習得のための勉強会	6
4.3	第 1 回保育園訪問	6
第 5 章	お助け!つくシフトについて	8
5.1	概要	8
5.2	要求分析	8
5.3	第 1 開発	10
5.3.1	到達目標	10
5.3.2	開発体制	10
5.3.3	開発内容	10
5.3.4	第 1 回納品	14
5.3.5	レビュー	14
5.4	第 2 開発	15
5.4.1	到達目標	15
5.4.2	開発体制	15
5.4.3	開発内容	15
5.4.4	第 2 回納品	16
5.4.5	レビュー	17
5.5	第 3 開発	17
5.5.1	到達目標	17
5.5.2	開発体制	17
5.5.3	開発内容	17
5.5.4	第 3 回納品	19
5.5.5	レビュー	19

5.6	第 4 開発	20
5.6.1	到達目標	20
5.6.2	開発体制	20
5.6.3	開発内容	20
5.6.4	第 4 回納品	21
第 6 章	つくポスについて	22
6.1	つくポスの概要	22
6.2	要求分析	22
6.3	到達目標	23
6.4	開発体制	23
6.5	開発内容	24
6.6	納品	25
第 7 章	成果発表会について	26
7.1	中間発表会	26
7.2	アカデミックリンク	26
7.3	最終成果発表会	26
第 8 章	使用技術	28
8.1	正規表現のライブラリ	28
8.2	勤務表の自動作成	28
8.3	郵便番号から住所検索	30
8.4	保育体制確認表自動出力	32
8.5	データベース	32
第 9 章	今後の課題と展望	33
9.1	つくシフト第4バージョンの納品	33
9.2	つくポスの納品	33
9.3	つくしの子保育園での運用	33
第 10 章	評価	34
10.1	活動内容に関する評価	34
10.2	開発内容に関する評価	34
10.2.1	つくシフトに対する評価	34
10.2.2	つくポスに対する評価	34
10.3	評価のまとめ	35
第 11 章	学び	36
11.1	タスク管理	36
11.2	情報の洗い出し	36
11.3	先方とのコミュニケーション	36
11.4	スケジュール管理	37
11.5	システムテストの重要性	37

11.6 ポスター作成	37
第 12 章 まとめ	38
付録 A 新規技術習得	39
付録 B 活用した講義	40
付録 C テスト項目	41
参考文献	42

第 1 章 はじめに

1.1 つくしの子保育園について

つくしの子保育園は函館市にある保育園である(図 1.1)。保育士 18 名園児定員 60 名が在籍しており、保育士は 0 歳から 5 歳までの園児の保育を行っている。一般的な保育に限らず、延長保育、産休明け保育、障害児保育、一時保育の様々な保育にも力を入れている。つくしの子保育園は市街地から離れており、静かな場所にある。建物は壁、床、天井が木で建築されており、ウサギや亀を飼育している。近所には馬、ロバ、ヤギなどを飼育している家もあり、様々な動物と身近に触れ合うことができる。保育園の傍には広い畑を借りている。秋には園児たちが育てた野菜を収穫し、調理をして食べる。このようにつくしの子保育園は自然に囲まれた環境にある[1]。



図 1.1 つくしの子保育園

(文責: 大谷笙互)

1.2 背景

1.2.1 保育業界の現状

現在、保育業界では保育士不足が問題となっている[2]。賃金が少ないと感じること、就業時間が希望に合わないことなど保育士不足の理由は様々ある。そして、保育士不足により新たに 2 つの問題が起きている。1 つ目は保育士 1 人 1 人の労働負担が大きくなっている問題である。主な仕事は園児たちが登園する前にその日することの準備、登園してからは在園児たちの世話、園児たちが退園してからは毎月あるお便りの作成や行事の準備など、非常に多忙である。2 つ目は受け入れ可能な園児の人数が減少してしまい、待機児童を増やすことに繋がってしまう問題である。待機児童は平成 28 年度で 23000 人を超えていた[3]。保育士不足を解消する政策を厚生労働

省はとっているが今だ解消し切れていないのが現状である [4].

(文責: 神秀樹)

1.3 つくしの子保育園における現状

現在のつくしの子保育園では、勤務シフト表の作成と卒園児のいる家庭へ送るはがきの作成を、手作業で行っており手間がかかっている。また、つくしの子保育園には事務員が1人しかおらず保育士が作業を手伝う場合も多々ある。しかし保育士は保育業務が主な仕事であるため、事務作業の手伝いを勤務時間外で行なっている。次の段落では、つくしの子保育園のシフト作成の業務の現状を述べ、続いての段落ではがき作成の業務の現状を述べる。初めに、勤務シフト表を作成する手順について述べる。保育士はまず保護者から園児1人1人の登園時間と退園時間の情報を紙媒体で受け取り、その情報をもとに保育体制確認表(図1.2)と呼ばれる時間帯ごとの園児の人数と、必要な保育士の人数をまとめた表を作成する。その後保育士の雇用形態などの情報と照らし合わせ勤務シフト表を作成することが手順となる。このとき、シフト作成に必要な条件が3つある。1つ目は園児ごとに異なる登園時間と退園時間に合わせた勤務シフト表を作る必要がある。2つ目は児童福祉法により、0歳児3人に対し保育士1人、1,2歳児6人に対し保育士が1人、3歳児20人に対し保育士1人、4,5歳児30人に対し保育士が1人[4]といったように決められている園児の年齢に対応する保育士の人数を守ること。3つ目は、保育士の労働時間を7時間15分以内に収める必要があることの3点である。勤務シフト表を作成する上で、こういった条件を守る必要がある。続いて、卒園児のいる家庭へ送るはがきの作成について述べる。現状のつくしの子保育園では卒園児のいる家庭へ行事の度にお知らせをはがきで送っている。はがき作成の手順は、卒園児の名前と住所がまとめられているWordファイルを印刷し、切り取り、はがきに1枚1枚糊付けするという手順である。このとき、はがき作成に必要な条件が2つある。まず1つ目は、兄弟のいる家庭に向けたはがきは連名で送ること。2つ目は年齢や行事によって送り先が変えること。はがきを送る機会は運動会、バザー、年賀状等と頻度が高い。したがって、保育士の負担になっているといえる。

2015年度 保育体制確認表 (4月)																
保母数	7:00	7:15	7:30	8:00	8:30	9:00	9:15	9:30	9:45	10:00	10:15	10:30	10:45	11:00		
ひよこ組	1	2	3	4	5	6	7	8	10	9	8	5	4	3	2	1
うさぎ組	1	2	3	4	5	6	7	8	9	10	9	8	7	6	5	4
かば組	1	2	3	4	5	6	7	8	9	10	10	9	9	5	5	2
ぞう組	6	12	13	18	18	18	18	18	18	13	12	8	8	3	0	
くじら組	1	10	3	10	6	8	10	10	10	10	10	10	8	8	1	1
いるか組	4	7	9	13	13	13	13	13	13	9	9	5	5	2	1	

図1.2 保育体制確認表

(文責: 神秀樹)

第 2 章 到達目標

2.1 活動目標

私たちは、プロジェクト活動する上で本プロジェクトの特徴として、クライアントがいるので、システムの納品予定の期日を守ることを目標とした。また、メンバー全員が様々な経験ができることでプロジェクトとしての学びも大きくなると考え、タスクを均等に割り振ることを目標とした。また、発表に関して、聴衆に伝わりやすいポスターを作ることや、発表技術の向上を目標とした。

(文責: 大谷笙互)

2.2 開発目標

私たちは、事務作業を簡略化できるようなシステムを開発することで、保育士の事務作業にかかる負担を軽減することを目標とした。開発サイクルでは、つくしの子保育園へのフィールドワークで得られた情報を元に、要求分析を行った。そうすることで保育士の方々でも使いやすいシステムを素早く開発し、保育園への納品の後に先方のレビューを参考にし、改善するというサイクルを繰り返すことを目標とした。そして、つくシフトに関しては、勤務シフト表の自動作成を可能とし、大幅な作業時間削減を実現することを目標とした。つくポスに関しては、卒園児のいる家庭に向けて送る、はがきの宛名面に卒園児の住所や氏名を直接印刷することを目標とした。

(文責: 黒瀬丈哉)

第3章 活動プロセス

3.1 年間スケジュール

私たちの活動スケジュールを以下に示す。

- 4月
 - プロジェクト発足
 - プロジェクトメンバーとリーダー決定
- 5月
 - グループ分け決定
 - グループリーダー決定
 - GitHub, Redmine, Cygwinなどのツールの導入
 - チームでGitの勉強会
 - フィールドワーク講座
 - 第1回保育園訪問
 - リーンキャンパスの作成
 - 設計
 - 開発開始
- 6月
 - 第1回月例レビュー会
 - チーム名を「保育エイド」に決定
 - システム名をお助け！つくシフトに決定
 - 第2回保育園訪問
 - 中間発表に向けたポスター作成
 - 発表ポスターレビュー会
- 7月
 - 中間成果発表会
 - 前期の振り返り
 - 中間報告書作成
 - 中間報告書提出
- 8月
 - 第3回保育園訪問
 - つくシフト第1バージョン納品
- 9月
 - プロジェクト後期活動の開始
 - チーム内で「つくシフト」「つくポス」のチーム分け
- 10月
 - つくシフト第2バージョンに追加する機能の要件定義
 - 5人でつくポスの実現方法について話し合い

Field Oriented System Design Learning by Users' Feedback

- プロジェクト全体で GitHub の勉強会
- 第 2 回月例レビュー会
- 11 月
 - アカデミックリンク
 - 第 4 回保育園訪問
 - つくシフト第 3 バージョンの納品
- 12 月
 - 最終成果発表会に向けたポスター作成
 - 発表ポスターレビュー会
 - 最終成果発表会
 - 後期活動の振り返り
- 1 月
 - 最終報告書の作成
 - 最終報告書の提出
 - 第 5 回保育園訪問
 - つくシフト第 4 バージョンの納品
 - つくポスの納品

(文責: 黒瀬丈哉)

第4章 開発への準備

4.1 開発に使用したツール

開発したアプリケーションのバージョン管理ツールとして, Git / GitHub を使用した. Git はソースコードやファイルの変更履歴を管理することができるシステムである. Git は変更履歴をリポジトリと呼ばれる場所に保存する. リポジトリには各メンバーの開発環境に作成するローカルリポジトリと, メンバーで変更履歴を共有するリモートリポジトリの 2 種類が存在するが, GitHub はこのうちリモートリポジトリを提供するサービスの 1 つである. 各メンバーの作業内容, すなわちソースコードの変更履歴をリモートリポジトリで共有, 統合することができるため, 複数のメンバーが同時に開発を進めることができる. タスク管理には, Redmine という Web アプリケーションを使用した. Redmine はオンライン上で利用できるオープンソースソフトウェアである. Redmine は「チケット」という機能を使いタスクを管理することで, ガントチャートやカレンダー, ロードマップなどで表示することができる. これにより, メンバーのタスクが今どのような状態なのかをチームで共有することが可能になった.

(文責: 渡邊隆広)

4.2 技術習得のための勉強会

私たちは Git / GitHub の勉強会を 2 度行なった. まず 6 月 1 日に大谷, 佐藤, 神の 3 名が勉強会に参加した. この勉強会では学部 4 年生の方を講師として招き, 各自でローカルリポジトリを作成し, pull を行うといったような基礎的な動きについて学んだ. 続いて, 10 月 12 日に本プロジェクトのメンバー全員で勉強会を行った. 産業技術大学院大学の中鉢欣秀様に講義を行っていただいた. 行った内容として, 実際にメンバー間でチームを組んでリモートリポジトリを共有し, チーム内でそれぞれファイルを GitHub に pull を行ったり, コンフリクションが起こった際の対処方法について学んだ.

(文責: 黒瀬丈哉)

4.3 第1回保育園訪問

私たちは活動に当たって事前に把握しているつくしの子保育園の情報を, メンバー間で共有し合った. そこから, システムの要件定義をするうえで必要となる詳しい情報を得るために, 保育園へ訪問することが決まった. 訪問日は 5 月 16 日となった. 第 1 回保育訪問は私たち 6 人に加え, 担当教員 3 名とティーチングアシスト(以降, TA とする) 5 名の計 14 名で行われた. 訪問時の録音データをドキュメントに書き出しを行った. 書き出しから大きく 2 つのことがわかった. 1 つ目は, はがき作成業務の改善より勤務シフト表の作成業務の改善の方が先方に望まれているということである. このことから私たちはつくシフトを前期で作成することを決めた. また, つくバスは後期から取り組むことに決めた. 2 つ目として, 実際に勤務シフト表を作成する上で困っている点だ. 困っ

ている点は以下の通りである。

- 園児の登園時間や退園時間によって、その時間に必要となる保育士の人数が変化すること
- 保育士の勤務時間は労働基準法に従うこと
- 保育士間で勤務時間の月合計ができるだけ均等にすること。
- 毎月 1 人で勤務シフト表の作成をしていること

これらの情報をもとに、私たちは要件定義、第 1 開発に取り組んだ。

(文責: 神秀樹)

第5章 お助け!つくシフトについて

5.1 概要

私たちが開発した Excel アプリケーションつくシフトは、つくしの子保育園における勤務シフト表の作成作業の負担軽減と、効率化を図るアプリケーションである。このアプリケーションの名称は、「つくしの子保育園」の勤務シフト表の作成作業を支援する、言い換えれば「助ける」というコンセプトから、その 2 つの単語を組み合わせたものである。使用用途は園児や保育士の登録から勤務表などの印刷までを想定している。既存の他のアプリケーションと比較した際の優位性として、「つくシフト」は「つくしの子保育園」に特化した自動シフト表作成システムであり、「つくしの子保育園」独自のルールに合わせた保育体制確認表の自動出力、既存の勤務表などの形を維持したまま編集できる点や、パソコンを使いなれない人が誰でも扱えるように「見やすさ・使いやすさ」を追求している点などがある。つくシフトは、まず園児と保育士の情報を登録する。園児の情報とは、氏名（姓、名）、登園時間、帰宅時間、誕生日、郵便番号（住所）、電話番号のことである。保育士の情報とは、氏名、雇用体系（正職員、パート、臨時職員）、担当クラスのことである。園児と保育士の情報の登録が終了した後、勤務表の月初めの 1 日の勤務を手動で入力する。手動で入力とは、事前に登録してある保育士情報を参照し、リスト形式で表示されている中からマウスでクリックするという作業のことを指す。月初めの 1 日の勤務の登録が終わった後は、自動作成ボタンをクリックすることで自動的にシフトが作成される。また、勤務を考える際は別シートにある「保育体制確認表シート」や「ガントチャートシート」を確認しながら作成することとなる。

（文責：渡邊隆広）

5.2 要求分析

私たちは、フィールドワークから得た情報や、先方からの要求を分析し、現状の課題や、解決策を考察した。分析するために用いたリーンキャンバスや、考察した課題と解決策を以下に記述する。

リーンキャンバス

リーンキャンバスとはビジネスモデル企画のためのツールの 1 つであり、新規サービスのアイデアを 9 種類の要素を用いて整理するためのテンプレートである。以下に要素の一覧を示す。

1. 課題、既存の代替品 (Problem, Existing Alternatives)
2. 独自の価値提案、わかりやすいコンセプト (Unique Value Proposition, High-level Concept)
3. ソリューション (Solution)
4. チャネル (Channels)
5. 収入の流れ (Revenue Streams)
6. コスト構造 (Cost Structure)
7. 主要指標 (Key Metrics)
8. 圧倒的な優位性 (Unfair Advantage)

Field Oriented System Design Learning by Users' Feedback

私たちはこの様式に則り各要素を検討することで、より綿密な要件定義を行うことができた。特に、私たちが作成するシステムは既存のシフト作成ツールと比べた際に以下の 3 点より優れていることを確認した。

- 費用がかからない。
- つくしの子保育園のシフト作成に特化している。
- 操作がとても単純である。

なお、今回のプロジェクトではビジネス化が目的ではないため、収入の流れやコスト構造についての検討は行わなかった。作成したリーンキャンバスを下の図 5.1 に記す。

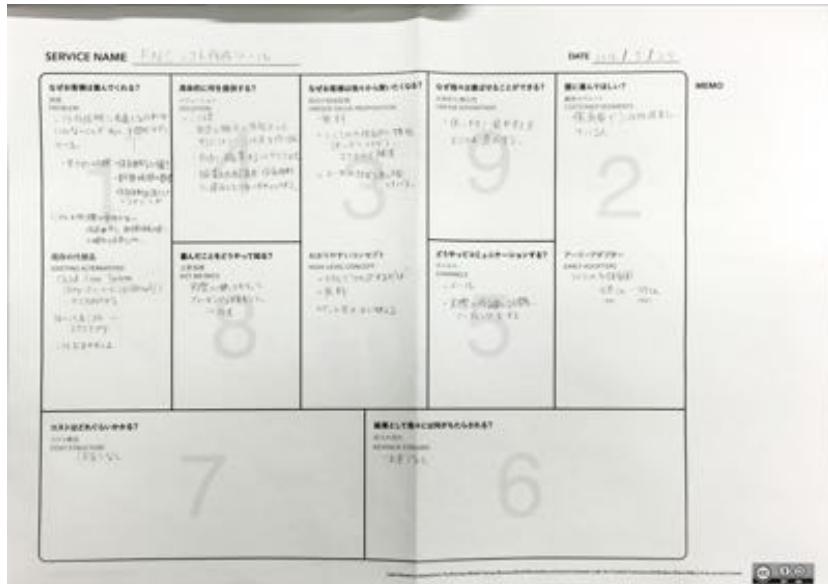


図 5.1 作成したリーンキャンバス

課題

- 先方は、Excel や Word の操作には慣れているものの、基本的に PC の操作に慣れていない。
- シフト作成に必要な園児の情報や保育士の情報の管理方法に統一性がなく、ファイル形式や保存フォルダが統一されていない。
- 現在先方が使用しているシフト表の形式を変更しない。
- システムで自動化できる部分とできない部分の線引きをし、あえて自動化しないほうがいい作業を判別する。
- 先方の PC のスペックが低く、処理が遅い。

解決策

- 先方が使い慣れている Excel で全て操作できるように、開発言語は Excel VBA を使用する。
- 園児の情報や保育士の情報は、Excel だけで管理できるようにし、管理方法を統一する。
- 保育体制確認表は、Excel で管理されている園児の情報を参照することで、自動で出力する。
- 勤務シフト表シフト表は簡単な操作で自動作成できるようにする。
- 勤務シフト表の勤務時間帯は時期によって例外的な勤務時間が発生するため、あえて自動化はせず、リストから選べるようにし、入力する手間を省く。
- なるべく処理が軽くなるようにコーディングする。

5.3 第 1 開発

5.3.1 到達目標

私たちは 5 月に行われた第 1 回訪問から得られた情報をもとに要求分析し、開発を行った。要求分析をした結果、勤務シフト表の生成のためには保育士の情報、保育体制確認表が必要であることがわかった。更に保育体制確認表には園児の情報が必要であった。現在でも保育士、保育園児それぞれの情報の管理はあったが、統一性がなかった。保育体制確認表においては、ファイルでの管理ではなく紙媒体での管理であった。このことから今回の開発では、勤務表作成に必要な情報である保育園児、保育士の情報の管理体制の一元化、保育体制確認表の自動集計、勤務表の生成、これらを目標とした。

(文責: 小嶋凌弥)

5.3.2 開発体制

第 1 開発では 4 人が実装に携わった。シフト表のレイアウト作成に 1 人、園児の情報登録、編集、削除機能にそれぞれ 1 人ずつ携わった。それぞれの機能が終わり次第、保育士の情報登録、編集、削除機能、保育体制確認表の自動算出を実装するものとした。

(文責: 小嶋凌弥)

5.3.3 開発内容

開発するシステムは、従来先方が Excel を用いてシフト作成を行っていたため、Excel の拡張が可能である Excel VBA を用いて実装を行った。また、Excel で開発するため、園児情報、保育士情報、保育体制確認表、勤務表それぞれをシートごとに分けて開発行った。第 1 開発においての主な機能の実装プロセスは以下の通りである。

- 園児情報シート

ここでは園児情報を管理するための以下 3 つの機能の実装を行った。園児をシートに登録する機能、編集する機能、削除する機能である。それぞれをボタンとしてシート上に置き、それらをクリックすることによって各作業を行えるようになっている。図 5.3.1 に園児情報登録画面を示す。「登録」ボタンをクリックすると「園児の名前」「園児の登園時間」「園児の退園時間」「園児の生年月日」の順にそれぞれ入力されるフォームが遷移し、すべての情報の入力が終わったら園児情報登録画面上のセルに出力される。この際、キーボードを使用して入力する情報は「園児の名前」のみであり、他の情報は選択するものとなっている。また、生年月日の情報をもとに「園児の年齢」と「園児の所属するクラス」を自動で計算し園児情報登録画面上のセルに出力される。「編集」ボタンをクリックすると「編集したい園児の名前」を入力するフォームが表示され、名前を入力後「編集したい情報」を選択するフォームへと遷移する。この際、「登録機能と同様に入力する情報は「園児の名前」のみであり、他の情報は選択ものである。「削除」ボタンをクリックすると「削除したい情報」を入力するフォーム

ムが表示され、入力後「本当に削除しますか？」という警告の後、任意の情報を削除する。この際、入力する情報に制限はなく、園児情報登録画面に表示されている文字列のすべてを検索し、一致する単語の行すべてを削除するものとなっている。

		登録	編集	削除	
名前	登園時間	帰宅時間	誕生日	年齢	クラス
satu	8:00	17:00	2012年4月4日	4歳	ぞう
aueo	7:00	17:30	2010年8月8日	5歳	いるか
いいいい	7:00	16:00	2015年5月3日	1歳	うさぎ
ああああ	7:00	15:30	2013年8月4日	2歳	ひよこ
園児U	7:00	16:00	2016年4月3日	0歳	ひよこ
園児C	7:00	18:00	2012年8月25日	3歳	くじら
園児Y	8:00	18:00	2016年2月6日	0歳	うさぎ
園児Z	8:00	16:30	2016年4月8日	0歳	ひよこ
園児I	8:00	17:30	2014年8月30日	1歳	かば
園児A	8:00	17:30	2015年3月5日	1歳	かば
園児R	8:30	17:00	2015年10月27日	0歳	うさぎ
園児V	8:30	16:30	2016年5月5日	0歳	ひよこ
園児X	8:30	17:00	2015年6月3日	1歳	うさぎ
園児S	8:30	16:00	2014年4月4日	2歳	かば
園児P	8:30	16:30	2012年8月23日	3歳	くじら
園児A	8:30	17:30	2012年4月6日	4歳	くじら
園児E	9:00	16:00	2015年8月27日	0歳	うさぎ
園児L	9:00	17:00	2015年4月2日	1歳	うさぎ
園児F	9:00	16:00	2013年3月18日	3歳	くじら
園児O	9:30	15:00	2016年5月20日	0歳	ひよこ
登録	9:30	17:30	2016年2月2日	0歳	うさぎ
園児AA	9:30	18:30	2015年5月17日	1歳	うさぎ
園児G	9:30	18:30	2014年5月29日	2歳	かば
園児H	10:00	18:30	2016年1月3日	0歳	うさぎ
園児D	10:00	16:30	2014年8月26日	1歳	かば

図 5.3.1 園児情報登録画面

- 保育士情報画面

保育士情報登録画面では、勤務表に必要な保育士情報を入力してもらい、それを表示する。画面上部には園児情報登録画面と同様に「登録」「編集」「削除」のボタンがあり、それらをクリックすることによって各作業を行えるようになっている。図 5.3.2 に保育士情報登録画面を示す。「登録」ボタンをクリックすると「保育士の名前」「保育士の雇用形態」「保育士の担当クラス」の順にそれぞれ入力するフォームが遷移しすべての情報の入力が終わったら保育士情報登録画面上のセルに出力される。この際、キーボードを使用して入力する情報は「保育士の名前」のみであり、他の情報は選択するものとなっている。「編集」ボタンをクリックすると「編集したい保育士の名前」を入力するフォームが表示され、名前を入力後「編集したい情報」を選択するフォームへと遷移する。「編集したい情報」を選択すると各情報を編集するフォームへと遷移する。この際、「登録」機能と同様に入力する情報は「保育士の名前」のみであり、他の情報は選択するものである。「削除」ボタンをクリックすると「削除したい情報」を入力するフォームが表示され、入力後「本当に削除しますか？」という警告の後任意の情報を削除する。この際、入力する情報に制限はなく、園児情報登録画面に表示されている文字列のすべてを検索し、一致する単語の行すべてを削除するものとなっている。

名前	渡邊	担当クラス
小嶋 凌弥	正職員	くじら
渡邊 隆広	正職員	かば
黒瀬 丈哉	パート	うさぎ
大谷 笙互	正職員	ぞう
神 秀樹	正職員	いるか
佐藤 恭祐	パート	ひよこ
山田 太郎	正職員	ひよこ
鈴木 一郎	正職員	うさぎ
高橋 淳平	正職員	かば
斎藤 隆司	正職員	ぞう
田中 康介	正職員	フリー
向井 和馬	パート	フリー
大西 雅也	正職員	くじら
三好 良弥	正職員	いるか
伊藤 泰斗	正職員	フリー
あ	パート	ひよこ

図 5.3.2 職員情報登録画面

- 保育体制確認表

保育体制確認表の自動出力機能は「園児情報登録画面」で入力した情報をもとに、時間帯ごとの園児の人数と必要な保育士の人数を自動で計算し出力する。また、クラス別で園児の人数と必要な保育士の人数を分けている。図 5.3.3 に保育体制確認表の画面を示す。

時間	7:00	8:00	8:30	9:00	9:30	10:00	10:30	11:00	11:30	12:00	12:30	13:00	13:30	14:00	14:30	15:00	15:30	16:00	16:30	17:00	17:30	18:00	18:30
保育士数	3	4	5	5	7	9	9	9	8	8	7	6	4	4	2								
ひよこ 4名	1	2	3	3	4	4	4	4	3	3	2	0	0	0	0								
担当保育士数																							
うさぎ 3名	0	1	3	5	7	8	8	8	8	7	7	4	3	3									
担当保育士数																							
かわ 7名	1	3	4	4	5	7	7	7	7	7	5	4	4	2	0								
担当保育士数																							
成山保育士数 残欠合計	2	6	10	12	16	19	19	19	18	18	14	11	8	5	3								
毛う 2名	0	0	0	0	0	1	2	2	2	2	2	2	1	0									
担当保育士数																							
くじら 5名	1	1	3	4	4	4	4	4	4	4	3	2	2	1	1								
担当保育士数																							
いるか 3名	0	0	1	2	2	3	3	3	3	2	2	1	0	0	0								
担当保育士数																							
成山保育士数 残欠合計	1	1	4	6	6	8	9	9	8	7	5	4	2	1									

図 5.3.3 保育体制確認表

• 勤務表画面

勤務表画面は既存の勤務表を模した形式で勤務時間帯、職員名を入力、編集、表示する。画面上部には西暦と月が表示されており、その下に平日のみの日付が「日付曜日」で複数表示される。西暦、または月を任意に変更することによって平日のみを自動で計算・出力し、格子なども自動で調整する仕様となっている。画面左部には勤務時間が表示されており、各セルをクリックすることで勤務時間のリストが表示され選択できるようになっている。リスト内の時間は15分きざみとなっており、コアタイムとなる時間は省いてある。画面中央部には保育士の名前が表示されており、各セルをクリックすることで保育士の名前のリストが表示され選択できるようになっている。リスト内の保育士の名前は保育士情報登録画面で入力した情報をもとに表示されており、苗字だけを表示するようになっている。図5.3.4に作成した勤務表画面を示す。

時間/日付		2016年8月																							
1月	2火	3水	4木	5金	6月	7火	8水	9金	10月	12火	15水	16木	17金	18月	19火	22水	23木	24金	25水	26木	29金	30火	31水		
7:00	~	15:00	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	佐藤	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	黒瀬		
7:15	~	10:30	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井	向井							
7:30	~	15:30	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	
8:00	~	16:00	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	
8:15	~	16:15	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	
8:30	~	16:30	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	
8:45	~	16:45	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	
9:00	~	17:00	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	
9:00	~	13:30	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤	伊藤							
9:15	~	17:15	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	
9:30	~	17:30	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	
10:00	~	18:00	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	高橋	青森	
10:15	~	18:15	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	青森	田中	
10:30	~	18:30	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好							
10:45	~	18:45	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好	三好							
10:55	~	18:55	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	
11:00	~	19:00	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	
11:15	~	19:15	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	
11:30	~	19:30	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	
11:45	~	19:45	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	
11:55	~	19:55	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	
12:00	~	20:00	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋
12:15	~	20:15	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤
12:30	~	20:30	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田
12:45	~	20:45	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木
12:55	~	20:55	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋
13:00	~	21:00	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森
13:15	~	21:15	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中
13:30	~	21:30	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋
13:45	~	21:45	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造
13:55	~	21:55	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬
14:00	~	22:00	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷
14:15	~	22:15	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神
14:30	~	22:30	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤
14:45	~	22:45	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田
14:55	~	22:55	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木
15:00	~	23:00	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋
15:15	~	23:15	青森	田中	小嶋	道造	黒瀬	大谷	神	佐藤	山田	鈴木	高橋	青森	田中	小嶋	道造								

5.3.4 第 1 回納品

2016 年 8 月 5 日（金）に 3 度目の保育園訪問を行い初めて「つくシフト」を納品した。第 1 回納品時点でのつくシフトの機能をシートごとに分けて以下に示す。

- 園児シート
 - 園児情報を登録、編集、削除する機能
- 保育士シート
 - 保育士情報を登録、編集、削除する機能
- 保育体制確認表シート
 - 園児情報を参照して保育体制確認表を自動的に表示する機能
- 勤務表シート納品時点ではつくシフトの説明と作業手順を実際に見せたのみであった。更につくシフトの説明書も事前にこちらで作成し、先方に渡して納品を終了とした。

(文責: 小嶋凌弥)

5.3.5 レビュー

納品時のつくシフトの評価は「従来管理していた園児の情報等をつくシフトに移すという手間があること」、「保育体制確認表を作成する作業が短縮された」となった。更に新たな要望として「1日の勤務の流れがわかるようなものがあれば良い」との声を頂いた。後は実際に使って頂いて何かあればグループリーダーのところまで連絡を頂くという方針にした。納品して数日後にグループリーダーのもとへ連絡が入った。「バグがあり、上手く動かない」との声を頂いた。バグの内容としては、クラスが変更されない、保育体制確認表が表示されないとのことだった。上記の評価をもとに後述する第 2 開発で取り組むこととした。実際にバグが起きた時のつくシフト画面を図 5.3.5 に示す。

	7:00	7:15	7:30	7:45	8:00	8:15	8:30	8:45	16:15	16:30	17:00	17:15	17:30	18:00	18:15
時間	7:00	7:15	7:30	7:45	8:00	8:15	8:30	8:45	16:15	16:30	17:00	17:15	17:30	18:00	18:15
保育士数	2	5	5	7	7	8	8	8	8	8	8	7	7	6	#N/A
ひよこ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ひよこ 0名															
担当保育士数	1	0	0	2	0	0	0	0	0	0	0	0	0	0	0
うさぎ	1	2	2	3	7	7	13	13	13	12	11	9	9	6	#N/A
#N/A															
担当保育士数	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
かわい	0	1	4	4	7	7	11	11	11	11	11	10	10	6	#N/A
#N/A															
担当保育士数	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
担当保育士数(園児合計)	1	1	3	4	6	2	7	14	5	14	5	24	5	24	5
ぞう	0	1	3	3	3	4	13	14	14	14	14	9	8	5	#N/A
#N/A															
担当保育士数	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0
くら	0	4	6	10	13	14	19	20	20	20	20	14	13	9	#N/A
#N/A															
担当保育士数	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
いるか	2	2	2	3	7	7	11	12	12	12	12	9	9	7	#N/A
#N/A															
担当保育士数(園児合計)	0	3	7	11	16	0	23	0	25	0	43	0	46	0	46

図 5.3.5 実際に起きたバグの画面

(文責: 小嶋凌弥)

5.4 第 2 開発

5.4.1 到達目標

第 2 開発の目標は、第 1 開発の納品の際に発生した、保育体制確認表の値が正しく入力されないバグとその他の細かいバグの修正や、勤務シフト表を自動作成する機能の実装、1 日ごとの勤務状態をガントチャートで表示する機能の実装をし、つくシフトの全ての機能を完成させ、実際に先方がつくシフトを使用して勤務シフト表を作成できるようにすることを目標とした。

(文責: 佐藤恭祐)

5.4.2 開発体制

第 2 開発からは、同時進行でつくポスの開発も始めたため、メンバーを 3 人ずつつくシフトチームとつくポスチームに分けて、それぞれのチームで開発を進めた。第 2 開発では、まず第 1 開発で発生したバグを修正した。その後担当を、第 1 開発で実装した園児情報シートの拡張と卒園児シートの作成、勤務シフト表の自動作成機能の実装、1 日ごとの勤務状況をガントチャートで表示させる機能の実装に 1 人ずつ振り分けて、開発を行った。

(文責: 佐藤恭祐)

5.4.3 開発内容

- 園児情報シートの拡張と卒園児シート

つくシフトがつくポスと連携して、つくポスで作成するはがきの送り先のデータとして、卒園児シートのデータを使用するために開発を行った。まず園児情報シートの登録する情報の項目に郵便番号と住所を追加し、その後卒園ボタンを作成した。卒園ボタンを押すと、自動で園児情報の中から卒園している年齢の園児を検索し、卒園している園児の情報を全て卒園児シートに移動させる。この卒園児シートに移動した園児の情報がつくポスのはがきの送り先のデータとして使用される。

- 勤務シフト表の自動作成機能

勤務シフト表の自動作成は、先方のシフト表の形式を変えずに自動作成できる。つくしの子保育園の勤務シフト表の特徴として、毎日同じ時間帯に勤務している職員と、他の時間帯をローテーションしている職員がいる。そのためつくシフトでも、この特徴に対応できるようにシステム設計をした。まず、毎日同じ時間に勤務する職員と他の時間帯をローテーションする職員を識別するために、勤務表を自動作成する前にその月の始めの日だけ勤務表を埋めてもらう仕様にし、勤務時間帯の左側に各行につき 1 つずつチェックボックスを作成した。このチェックボックスに、毎日同じ時間に勤務する職員がその行に入力されている場合はチェックを付け、そうでない場合はチェックを付けないようにすることで、毎日同じ時間に勤務する職員とそうでない職員を識別する。次に勤務シフト表を自動作成するために、自動作成ボタンを設置した。このボタンは、その月のはじめの日だけ勤務表を埋めて、チェックボックスにチェックを入れ終わったら押す。そうすると、チェックをつけた行は毎日同じ職員が入り、チェックを付けていない行はそのほかの職員でローテーションするように自動

Field Oriented System Design Learning by Users' Feedback

で勤務シフト表が埋まる。第 1 開発と同じで勤務時間帯は手動で入力する仕様にしている。作成した勤務表画面を図 5.4.1 に示す。

図 5.4.1 勤務表画面

- ガントチャート

新たにガントチャートシートを作成し、そのシートに機能を実装した。ガントチャートは勤務表シートと連動し、どの職員がどの時間帯に勤務しているかを横棒グラフで表したものである。グラフの横軸は時間、縦軸は職員の名前で、職員が勤務している時間帯に棒グラフが表示される。そしてこのグラフの枠の上側に勤務シフト表 1 ヶ月分の日付が表示されていて、クリックするとその日付のグラフを表示させる。作成したガントチャートを図 5.4.2 に示す。

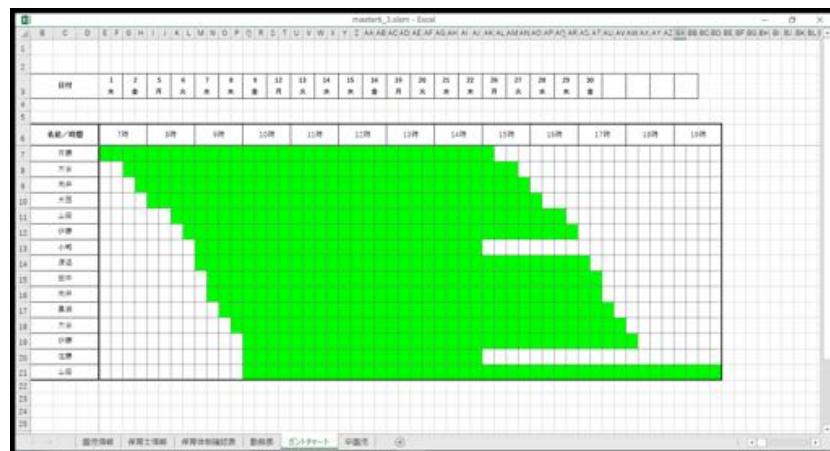


図 5.4.2 ガントチャート

(文責: 佐藤恭祐)

5.4.4 第 2 回納品

2016 年 10 月 20 日 (木) に第 3 回保育園訪問を行い、つくシフトを納品した。あらかじめ USB メモリにつくシフトのデータを入れたおき、USB メモリから先方のパソコンにデータを移した。その後、第 1 開発で発生したバグが修正されているか確認し、一通り勤務表自動作成までの操作を行い、バグを探した。特に目立ったバグはなかった。そして、先方に実際に勤務表の自動作成の操作をしていただき、ガントチャートも確認していただいた。その後先方からレビューをいただいた。

(文責: 佐藤恭祐)

5.4.5 レビュー

実際に第2開発で改良したつくシフトを先方に使用していただき、以下のレビューをいただいた。

- 勤務表自動作成

勤務表自動作成についての評価はとても高く、勤務表を作成する手間が省けることをとても喜んでいただけた。しかし、先方のPCで実行したところ、処理速度が非常に遅く、自動作成に3分ほどかかってしまった。処理速度に対しては改善してもらいたいという要望をいただいた。さらに、勤務表の途中の日付からでも自動作成を可能にして欲しいという要望もいただいた。

- ガントチャート

ガントチャートについての評価も高く、1日ごとの職員の勤務状態が一目でわかるようになり、とても喜んでいただけた。

(文責: 佐藤恭祐)

5.5 第3開発

5.5.1 到達目標

第3開発では「勤務表自動作成機能の完成と処理速度改善」「園児情報シートの登録ボタンを押した際に表示されるダイアログの改良」「ガントチャートシートの改良」「保育士情報シートの登録方法を微調整」「勤務表シートにクリアボタンとチェックOFFボタンの追加」「『つくポス』と連携するための調整」を到達目標とした。

(文責: 渡邊隆広)

5.5.2 開発体制

第3開発は第2開発時点での体制と同じく3人で開発を行った。作業の分担は、「ガントチャートシートの改良」「保育士シートの登録方法を微調整」「勤務表自動作成機能の処理速度改善」に1人、「園児シートの登録ボタンを押した際に表示されるダイアログの改良」に1人、「園児シートの登録ボタンを押した際に表示されるダイアログの改良」「つくポスと連携するための調整」に1人で開発の分担した。また、「勤務表自動作成機能の完成」は2人で行った。

(文責: 渡邊隆広)

5.5.3 開発内容

- 勤務表自動作成機能の完成と処理速度改善

第2回納品の時点では、勤務表を自動的に作成することはできたが「固定勤務のチェックボックスが反映されない」「自動作成時にうまくローテーションされていない」などのバグ

があったため、そのバグを修正し問題なく動作するよう機能を完成させた。また先方の PC 上で動作確認をした結果、作成ボタンを押してから処理が終了するまで平均 3 分程度かかっていた。先述した通り、処理速度を改善してほしいと先方から要望があったため、スペックの低い PC でも高速に動作するようアルゴリズムの改良を行なった。

- 園児情報シートの登録ボタンを押した際に表示されるダイアログの改良

第 2 回納品の時点では登録ボタンを押した後、「園児の名前の登録」→「登園時間」→「帰宅時間」…といったように 1 つの情報につき 1 つの入力フォームを表示していたが、これでは画面遷移が多くなってしまい使用者のストレスに繋がる可能性があると判断したため改良した。具体的には、画面遷移をなくして 1 つの入力フォームで全ての情報が入力できるようにした。改良したダイアログを図 5.5.1 に示す。



図 5.5.1 改良したダイアログ

- ガントチャートシートの改良

第 2 回納品の時点では、ガントチャートの上の日付をクリックするたびに勤務時間を参照し直しガントチャートを再表示していたが、無駄な処理であることとクリックするたびに再表示されることが使用者のストレスに繋がる可能性があると判断したため改良した。日付をクリックするたびに参照し直し再表示するのではなく、シートの切り替えごとに処理を行うようにした。

- 保育士情報シートの登録方法を微調整

第 2 回納品の時点では保育士登録の際、氏名を姓と名に分けて登録する仕様にしていたが、従来の勤務表は自由な名前で作られていたため、姓名で分けて入力する仕様を変更し、自由な名前で登録できるようにした。

- 勤務表シートにクリアボタンとチェック OFF ボタンの追加

第 2 回納品の時点では、勤務表を自動作成する際に範囲内のセルを空白にする作業を使用者が選択して削除する形をとっていたが、クリアボタンを追加することでその作業の負担を軽減した。また固定勤務のチェックボックスを外す際、複数のチェックが入っていた時も 1 つずつクリックしなければいけなかったものを、一括で OFF にできるボタンを追加し負担を軽減した。

- つくポスと連携するための調整

つくポスと連携する際に卒園児シートのデータを参照するため、卒園児の情報を CSV ファイルに変換する機能が必要となった。また CSV ファイルに変換する際に、卒園児の中に兄弟がいた場合は 1 つにまとめる処理を行う必要があったため、兄弟は住所が同じかどうかで判

定し 1 つにまとめた。加えて、CSV ファイルを「つくポス」側から読み込む際に卒園児の情報の順序を調整する必要があったため、並び替えを変換の際に行うようにした。

(文責: 渡邊隆広)

5.5.4 第 3 回納品

2016 年 11 月 15 日（火）に 4 度目の保育園訪問を行い「つくシフト」を納品した。第 3 回納品時点でのつくシフトの機能をシートごとに分けて以下に示す。

- 園児シート
 - 園児情報を登録、編集、削除する機能
 - 卒園児シートに園児情報を移動する機能
 - 園児情報を項目ごとにソートする機能
- 保育士シート
 - 保育士情報を登録、編集、削除する機能
- 保育体制確認表シート
 - 園児情報を参照して保育体制確認表を自動的に表示する機能
- 勤務表シート
 - 勤務表を自動作成する機能
 - 一括で固定勤務を判定するチェックボックスを外す機能
 - 一括で勤務表内の情報を削除する機能
- ガントチャートシート
 - 勤務表の情報を参照してガントチャートを表示する機能
 - シフトを日付ごとに表示する機能
- 卒園児シート
 - 卒園児の情報を項目ごとにソートする機能

また、納品の際にはマニュアルも納品し内容を先方と一通り確認した後、実際に「つくシフト」を使ってシフト作成の流れを掴んでもらった。園児や保育士の情報登録を行い、先方からのレビューをいただき納品の完了とした。

(文責: 渡邊隆広)

5.5.5 レビュー

納品をした際、先方に第 3 回納品時点での「つくシフト」を評価していただいた結果、「自動作成が早くなった」「頑張って使ってみます」などの声をいただいた。また、新たな要望として「勤務表を職員の人数の変動に対応できるようにしてほしい」「園児情報の住所を登録する際、郵便番号から住所を自動的に補完できるようにしてほしい」「卒園する際に渡す証書に在園日数を記入するので、それを自動的に計算し表示できるようにしてほしい」「勤務表自動作成機能で途中の日付からも自動作成できるようにしてほしい」といった要望を伺った。これは後述する第 4 開発で取り掛かることとした。

5.6 第 4 開発

5.6.1 到達目標

第 4 開発では第 3 回納品時に先方から頂いた要求、私たちからの先方に対しての提案をもとに開発を行った。要求内容としては「シフト表を職員の人数の増加に対応」、「園児登録時に郵便番号から住所を自動で補完」、「園児の入園日から今日までの在園日数の自動出力」、「途中の日付から勤務表自動作成」である。私たちからの提案としては「印刷ボタンの追加」、「保育体制確認表のユーザインターフェース改善」である。これらを到達目標とした。

(文責: 小嶋凌弥)

5.6.2 開発体制

第 4 開発は第 2、第 3 開発時点での体制と同じく 3 人で開発を行った。作業の分担は「シフト表を職員の人数の増減に対応」、「印刷ボタンの追加」、「保育体制確認表のユーザインターフェース改善」に 1 人、「園児登録時に郵便番号から住所を自動で補完」、「園児の入園日から今日までの在園日数の自動出力」に 1 人、「途中の日付から勤務表自動作成」に 1 人で行った。

(文責: 小嶋凌弥)

5.6.3 開発内容

- シフト表を職員の人数の増加に対応

保育士の人数の減少には対応していたが、増加に対応はしていなかった。これまでシフト表の保育士が登録できる人数は最大 15 人であったが、最大 20 人と増加した。

- 園児登録時に郵便番号から住所を自動で補完

これまで郵便番号、住所をそれぞれ入力していただく仕様となっていた。しかし第 3 回納品時に「郵便番号を入力すると住所が自動で出てきてほしい」との要望をいただいた。郵便番号入力欄の横に「住所検索」ボタンを設けた。郵便番号を入力し終わってから「住所検索」ボタンをクリックしてもらうことで、次の欄である住所入力欄が自動で補完されることとなるように実装を行った。今回の実装はアドインファイルで行うのではなく、Web サービス API から用いることとした。詳しい技術内容は使用技術にて後述する。

- 園児の入園日から今日までの在園日数の自動出力

これまで園児を登録したと同時にその登録した日付を入力するだけのものだった。しかし、先方からの要望により、登録日付から今日までの在園日数を出力するものとした。登録日付は Date 関数により園児登録を行った日付が登録日付となる。更に Today 関数を用いた。Today 関数は今日の日付を表示するものである。Today 関数から Date 関数を引くことにより、今日までのその園児の在園日数がわかるようになるものとした。表現方法としては「○年○ヶ月経過しました。」とするようにした。登録した日は「0 年 0 ヶ月経過しました。」となるが、1 ヶ月経つごとに自動更新するようになっている。

Field Oriented System Design Learning by Users' Feedback

- 途中の日付から勤務表自動作成

これまででは 1 日からしか自動作成を行うことができない仕様であった。しかし、第 3 回納品時に先方からの要望で「途中の日付から自動作成はできないか」と頂いた。変更したい日付のシフトにカーソルを合わせた状態で「作成」ボタンを押すとその日付から生成できるようにした。この仕様改善に伴い、1 日から生成する場合も 1 日にカーソルを合わせてから「作成」ボタンを押すようにしなければならなくなつた。

- 印刷ボタンの追加

これまで保育体制確認表と勤務表の印刷ボタンがなかった。しかし、従来は紙媒体で保育体制確認票と勤務表がつくしの子保育園に置いてあった。Excel にも印刷ボタンはあるが、先方にもわかりやすく、すぐに印刷できるようにとそれぞれのシート上に印刷ボタンの追加を実装した。更に保育体制確認表の増減、勤務表の職員の増減に対応してサイズを変えるようしている。

- 保育体制確認表のユーザインターフェース改善

これまでの保育体制確認表は従来の保育体制確認表を真似て、数字を表示しているだけであった。しかし、クラスをまとめた場合の担当保育士の人数は、クラスごとの担当保育士の人数を表示させる欄とは別の欄に表示させ、クラスごとの担当保育士の人数は空白を表示させた。そして空白になった担当保育士の欄は緑色で塗りつぶし、クラスをまとめていることが一目でわかるようにした。図 5.6.1 に改善した保育体制確認表を示す。

図 5.6.1 改善した保育体制確認表

(文責: 小嶋凌弥)

5.6.4 第4回納品

第4回納品に関しては9章にて後述する。

(文責: 小嶋凌弥)

第 6 章 つくポスについて

6.1 つくポスの概要

私たちが開発したはがき作成支援システム「つくポス」は、つくしの子保育園におけるはがきの作成作業の負担軽減と、効率化を図るシステムである。このシステムの名称は、つくしの子保育園のはがきの作成を支援するというコンセプトから、つくしの子保育園とポストカードを組み合わせたものである。使用言語は主に、HTML, CSS, JavaScript である。使用用途はつくシフトの卒園児情報をもとに、はがきを送る園児の選択と宛名の印刷までを想定している。既存の他のシステムと比較した際の優位性として、つくポスはつくしの子保育園に特化したはがき作成支援システムであり、つくシフトをデータベースとして園児の情報を読み込むことで、新たに住所情報を入れる必要がない点や送り先の選択をつくしの子保育園の独自のルールで行える点である。つくポスは、まず Internet Explorer でつくポスを開く。次に「はがきの宛名を印刷する」か「園児の情報を閲覧する」を選択する。この時、「園児の情報を閲覧する」を選択するとつくシフトが開かれる。「はがきの宛名を印刷する」を選択すると「おたよりはこちら」か「年賀状はどちら」のどちらかを選択する画面に遷移する。つくしの子保育園では、はがきの送り先に兄弟がいた場合、おたよりは兄弟を連名にして、年賀状は連名にしない。「おたよりはこちら」を選択すると兄弟がいた場合連名の判定を自動的に行う。「年賀状はどちら」を選択すると兄弟がいても 1 人 1 人に送るため連名の判定は行われない。その後、送り先の選択をして CSV ファイルとして保存する。送り先の選択では、つくしの子保育園では行事によって函館市近郊に送る場合があるため「函館市近郊ボタン」を作りボタン 1 つで送り先の選択を可能にした。また、年賀状では卒園児の学年によってはがきの内容を変えているため、「学年を選択する」ボタンを作り選択した学年を一括で送り先に指定できるようにした。保存をした後、ファイルを選択できるところで先ほど保存した CSV ファイルを選択して「印刷」ボタンをクリックする。最後に、開かれた Word でそのまま印刷を開始する。

(文責: 神秀樹)

6.2 要求分析

- 訪問に向けて

我々は、つくポスを作成するにあたって、つくしの子保育園を訪問して、はがき作成を支援する上でどのような要求があるのか、またつくしの子保育園のパソコン環境やプリンター環境などの制約条件について調査することになった。後期の最初の活動が 9 月 21 日であったのに対し、つくしの子保育園側と我々の都合により、訪問日は 10 月 21 日に決まった。訪問日まで約 1 ヶ月とだったので、訪問日までにわかる範囲でシステムの実現方法について検討した。その際、第 1 回保育園訪問でわかったことを参考にした。まず、候補として挙がったのがはがきでイベントの告知をしていたものをメールで送信するという案である。概要としては、QR コード をはがきに添付し、保護者に送付する。そこから保護者に情報を登録してもらうことで、保護者の氏名や住所などを登録したデータベースを作成し、データベースに登録されている保護者にメールを一斉送信するというものであった。しかし、この案は我々のミー

ティングで却下することになった。理由は、QR コードを作成する際、仮に卒園児 1 人 1 人に QR コードを作成するとした場合、本プロジェクトが終了した後の継続可能性が低いということ、QR コードを 1 種類のみ使った場合、入力しなければいけない項目が増え、保護者側の負担が増えるということである。以上より、この方法は有用ではないと考えた。そこで、新たに考案した案ははがき作成のフローを簡略化できるようなシステムを開発するという案である。現在のつくしの子保育園でのはがき作成のフローを調査し、簡略化できる部分についてシステム化しようという考えである。これを実現するためには、より詳細にはがき作成の手順について知る必要があると考え、訪問時に質問することになった。

- 保育園訪問

10 月 20 日木曜日につくしの子保育園を訪問し、はがき作成の手順について中心にお話をうかがった。この訪問でわかったことを以下にまとめる。

- はがきの宛名面は印刷したものを貼り付けている。
- 卒園児の情報は Word ファイルにて管理している。
- 過去のはがきの内容を編集して作成している。
- 年賀状を作る際、宛名面は兄弟で連名にしない。
- はがきの送り先は函館市近郊にのみ送っている場合や、現在の学年によって送る文面が変わることがある。
- 年賀状は卒園児が中学生を卒業するまで送っている。
- お便りは卒園児が小学校卒業まで送っている。

これを基にシステムに実装する機能を以下のように定めた。

- 過去に作成したはがきの文面を参考に編集しやすいような機能
- 学年や地域で送り先を絞り込みする機能
- その他、自由に送るかどうかの選択をする機能

この 3 点を念頭に開発に取り掛かることに決めた。

(文責: 大谷笙互)

6.3 到達目標

勤務時間外に行われている宛名の切り取り、糊の貼り付けの作業をつくポスを作成することによって無くし、保育士の負担を減らすこと。またそのシステムを使いやすいユーザインターフェースに整えることを到達目標とした。

(文責: 神秀樹)

6.4 開発体制

チームの 3 人が開発に取り組んだ。それぞれの役割として、2 人が「はがきの宛名の面を印刷する機能」に取り組み、他の 1 人が「はがきの文面を印刷する機能」に取り組んだ。

(文責: 黒瀬丈哉)

6.5 開発内容

- ホーム画面

システムのホーム画面にはボタンが 2 つある。まず「はがきを印刷する」ボタンをクリックすると、はがきの宛名を印刷するための画面へ遷移される。また、「園児の情報を閲覧する」ボタンをクリックすると、つくシフトが開き、卒園児の情報の閲覧をすることができる。図 6.5.1 に作成したつくポスのホーム画面を示す。



図 6.5.1 つくポス、ホーム画面

- 「お便り or 年賀状」選択画面

ホーム画面の「はがきを印刷する」ボタンをクリックすると、この画面に遷移する。この画面では、運動会やバザーなどのおたよりの作成に使用する「おたよりはこちら」ボタンと、年賀状の作成に使用する「年賀状はこちら」ボタンの 2 つのボタンが実装されている。主な違いとして、はがきに兄弟などの連名が必要な場合におたよりの作成を行い、はがきに連名が必要ではない場合に年賀状の作成が行われる。図 6.5.2 に作成した選択画面を示す。



図 6.5.2 「お便り or 年賀状」選択画面

- 宛名印刷画面

この画面ではつくシフトの卒園児シートのデータが表示されている。「印刷する」という文字の下にあるデータが印刷される。また、「函館市近郊にのみ送る」というボタンをクリックすると、函館市や北斗市、七飯町森町の 4 つの市町の卒園児のみを絞り込むことができる。そして、卒園児の学年によっても絞り込みが可能であり、「学年を選択する」ボタンをクリックすると、小学 1 ~ 6 年生と中学 1 ~ 3 年生のラジオボタンが表示され、このボタンをクリックすることで学年による印刷の有無選択が可能である。印刷するデータが定まったら、

Field Oriented System Design Learning by Users' Feedback

保存ボタンをクリックする。すると園児宛名情報 CSV という CSV ファイルが保存される。この CSV ファイルを参照し、印刷ボタンを押すことで Microsoft Word が立ち上がり、印刷することが可能になる。図 6.5.3 に作成した送り先選択画面、図 6.5.4 に立ち上がった Word ファイルを示す。



図 6.5.3 送り先選択画面

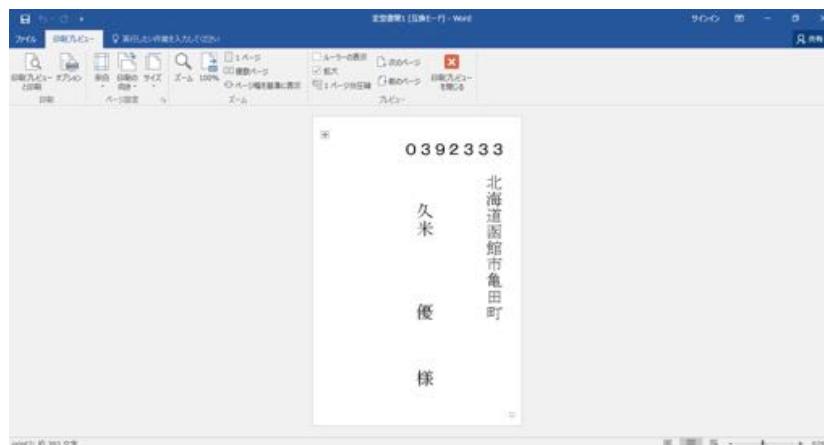


図 6.5.4 立ち上がった Word ファイル

(文責: 黒瀬丈哉)

6.6 納品

納品に関しては 9 章で後述する。

(文責: 神秀樹)

第 7 章 成果発表会について

本章では、これまで参加した成果発表会に向けての活動や、そこで得られたものについて述べる。

7.1 中間発表会

7月8日、公立はこだて未来大学で中間成果発表会が開催された。発表の対象は他のプロジェクトの学生や、教員であった。発表にはA1サイズのポスターを作成し、ディスプレイに作成中のシステムを写し、発表を行った。1回の発表の流れは、プロジェクト全体の概要説明（1分）、グループ毎に発表（7分）、質疑応答（7分）とした。発表を聞きに来た方にはアンケートに答えていただいた。項目は、発表技術について、発表内容について、システムの改善点についての3点であった。評価としては、システムについては先方のニーズを十分に理解できていると、一定の評価を得ることができた。しかし、ユーザインターフェースが不十分であることや、発表の時間が長いなどの評価もあった。これを受け、今後は実装予定している機能に加えて、ユーザインターフェース面での検討も必要であることがわかった。また、今後の発表会では発表時間についても検討することにした。

（文責：大谷笙互）

7.2 アカデミックリンク

11月12日、函館市青年センターでHAKODATEアカデミックリンクが開催された。会場にいらっしゃったお客様や、審査員、他のブースで発表している方に向けて発表を行った。発表では、A1ポスターを作成し、ディスプレイに作成したシステムを写しながら行った。また、ポスターセッションの形式を採用し、見に来ていただいた方とインタラクティブにやり取りができるようにした。HAKODATEアカデミックリンクでは論理性、想像力・発想、発表の工夫、熱意・努力、密着性の5つの観点で審査が行われるが、本グループは受賞には至らなかった。また、今回の発表でいただいたコメントでは、つくしの子保育園以外の保育園や、他の企業でも応用できるとなお良いというコメントをいただいた。これについては、今回のプロジェクト活動ではシステムとして改良を施すことはスケジュールの面で取り組まないことにしたが、自動で勤務シフトが作成されることについての需要の高さを感じることになった。

（文責：大谷笙互）

7.3 最終成果発表会

12月9日、公立はこだて未来大学で最終成果発表会が開催された。発表の対象は中間成果発表会と同様、学生と教員である。発表にはA1のポスター2枚とディスプレイを用意し、ポスターとシステムのデモを交えながら発表した。今回は、全体の概要を説明する時間ではなく、グループごとに行った。本グループでは、説明とデモに10分と質疑応答に5分の計15分を1回の発表の流れに設定した。作成したシステムが「お助け！つくシフト」と「つくポス」の2つだったので、聞き

Field Oriented System Design Learning by Users' Feedback

に来た方が迷わないように、つくシフトの発表を先に行い、質疑応答まで行ってから、つくポスの発表をするように発表の流れを工夫した。また、今回もアンケートを行い、作成したシステムがつくしの子保育園の現状を解決するのに十分であるかどうか、10点満点の点数をつけていただいた。平均点は8.5点であったので、我々は目標通りのシステムを作成できたと評価する。

(文責: 大谷笙互)

第 8 章 使用技術

8.1 正規表現のライブラリ

園児情報登録する際に郵便番号、電話番号を登録する場所がある。その際に誤った表記がないように正規表現を用いて制御するようにした。郵便番号においては「半角数字 3 桁 + ハイフン + 半角数字 4 桁」になっているかをチェックした。半角ではない数字の使用、桁数違い、ハイフンの位置違いはエラーを起こすようにしてある。電話番号はすべて半角で市外局番が 0 始まりで 2 桁から 5 桁、市内局番は 1 ~ 4 桁、後の電話番号が 4 桁となっているかをチェックした。今回は VBScript RegExp オブジェクトを使用し、参照設定は Microsoft VBScript Regular Expressions 5.5 とした。しかし、Windows にしかこの参照がないため、Mac には対応していないのが難点である。以下、正規表現を使用したうえで主となるソースコードを記す。

Option Explicit

```
Private Const g_cnsTitle As String = "住所録新規登録"  
Private Const g_cnsBlank As String = ""
```

，郵便番号用正規表現パターン

```
Private Const g_cnsRegPatternZIPCODE As String = "^[0-9]{3}-[0-9]{4}$"
```

、電話番号用正規表現パターン（国内のみ）

```
Private Const g_cnsRegPatternTELNO As String = "^\d{1,4}[-]\d{1,4}[-]\d{1,4}[-]\d{1,4}$"
```

、全角英数カナ用正規表現パターン

(文責: 小嶋凌弥)

8.2 勤務表の自動作成

勤務表の自動作成機能はシート内にある「作成」ボタンを押すことで自動的に勤務表を作成する機能である。自動作成に必要な準備として以下の条件がある。

- 基本となる 1 日のシフトが勤務表内に入力されていること
 - 自動作成する勤務表の範囲内が空白になっていること
 - 基本としたいシフトが入力されている列のセルをどれか選択していること

以上の条件を満たしていれば勤務表を自動作成することができる。また、「作成」ボタンを押す前に勤務表左側にあるチェックボックスにチェックを入れると、その行は固定となり自動作成時に変化しなくなる。以下に作成したソースコードを記述し、その内容について記述する。

固定勤務かどうかを判定する関数

```
Function isFixed(ByVal raw, col_left,col_right As Long) As Boolean
    Dim sht As Worksheet
    Set sht = Worksheets("勤務表")
    If sht.Cells(raw, col_left).Value <> sht.Cells(raw, col_right + 1) _ 
        And sht.Cells(raw, col_left).Value <> sht.Cells(raw, (col_right + 1) / 2) Then
        isFixed = True
    Else
        isFixed = False
    End If
End Function
```

ローテーションする人を List に格納する関数

```
Sub getChangeChildminderList(ByVal ChangeChildminderList As Collection, ByVal row_bottom, co
    Dim i As Long
    Dim sht As Worksheet
    Set sht = Worksheets("勤務表")

    ' 動く勤務の人たちをリストに追加するループ
    For i = 4 To row_bottom Step 1

        ' 固定かどうか判定
        If isFixed(i, col_left, col_right + 1) Then

            ' 固定じゃない人だけリストに追加する
            ChangeChildminderList.Add sht.Cells(i, col_left).Value
        End If
    Next

End Sub
```

勤務表を作成する関数

```
Sub CreateWorkSchedule(ByVal ChangeChildminderList As Collection, ByVal row_bottom, col_left
    Dim num_raw As Long: num_raw = 1
    Dim num_col As Long: num_col = 1
    Dim sht As Worksheet
    Set sht = Worksheets("勤務表")

    ' 勤務表埋めるループ（縦のループ）
    For i = 4 To row_bottom Step 1

        ' まず固定かどうか判定する
    Next
```

```
If isFixed(i, col_left, col_right + 1) Then  
  
    , 固定じゃないときに回す横のループ  
    For j = col_left + 1 To col_right + 1 Step 1  
  
        , ここの if 文でリストの中身をループさせてる（横に埋めるとき用）  
        If num_col = ChangeChildminderList.count Then  
  
            , リストの番号が最後だったときは最初にする  
            num_col = 1  
        Else  
  
            , リストの番号が最後以外のときは次の番号に  
            num_col = num_col + 1  
        End If  
  
        , 各セルに値を代入する  
        sht.Cells(i, j).Value = ChangeChildminderList(num_col)  
    Next  
  
    , ここでもリストの中身をループさせる（縦に移動したとき用）  
    If num_raw = ChangeChildminderList.count Then  
        num_raw = 1  
    Else  
        num_raw = num_raw + 1  
    End If  
  
    , 横で回した値を縦で回す値に置き換える  
    num_col = num_raw  
  
End If  
Next  
  
End Sub
```

（文責: 渡邊隆広）

8.3 郵便番号から住所検索

園児情報登録する際に郵便番号を入力し、園児情報登録画面上に置いてある「住所検索」ボタンを押すと、次の住所入力欄が自動的に埋まるようになっている。Microsoft が公式に配布しているアドインもあるが、先方の PC 環境をこちらからできるだけ変えないというのが条件にもあったた

Field Oriented System Design Learning by Users' Feedback

め、Web サービス API [5] を用いて実装することとした。しかし、これはインターネットに常時繋がっている環境でないと動作しないという点も併せ持っているが先方はインターネット環境があるという点と PC は保育園の中でしか使用しないという点から以上のこととは特に問題がないと判断した。以下に主となるソースコードを記す。

```
Function ZipCodeToAddress(strZipcode)

Dim objXMLHttp As Object, zipArr

'--ハイフンが入っていた場合は取り除く

strZipcode = Replace(strZipcode, "-", "")

Set objXMLHttp = CreateObject("MSXML2.XMLHTTP")

objXMLHttp.Open "GET", "http://zip.cgi.biz/CSV/zip.php?zn=" & strZipcode, False

objXMLHttp.Send

'API の結果を配列に代入する

zipArr = Split(Replace(objXMLHttp.responseText, """", ""), ", ")

'正常な値が返ってきた場合は配列の要素数が 15 になる

If UBound(zipArr) = 15 Then

    ZipCodeToAddress = zipArr(12) & zipArr(13) & zipArr(14)

Else

    '郵便番号が間違っている場合や未入力の場合は、空文字を返す

    ZipCodeToAddress = ""

End Function
```

End If

End Function

(文責: 小嶋凌弥)

8.4 保育体制確認表自動出力

保育体制確認表は、Excel の関数とマクロを組み合わせて実装した。形式は先方が使用しているものとほとんど変えずに自動出力できるようにした。自動出力は園児情報シートから Excel の関数を用いて、園児の「登園時間」、「帰宅時間」、「所属クラス」を参照し、時間ごとのクラス別の園児の人数や、クラスごとの園児の合計人数を自動で集計する。保育体制確認表の外枠は、Excel の機能の 1 つである条件つき書式を用いて、集計したデータの量によって自動で拡大縮小される。その後、マクロを用いてクラスごとの時間別に必要な保育士の人数を計算し、出力する。必要な保育士の人数は、園児の年齢に応じた必要な保育士の人数が決められている児童福祉法を基に計算している。具体的に挙げると、0 歳児 3 人につき保育士最低 1 人、1 歳児 5 人につき最低保育士 1 人、2 歳児 6 人につき保育士 1 人、3 歳児 15 人につき保育士最低 1 人、4 歳児と 5 歳児は 20 人につき保育士最低 1 人必要、というように決められている。つくしの子保育園ではひよこ組が 0 歳児クラス、うさぎ組が 1 歳児クラス、かば組が 2 歳児クラス、ぞう組が 3 歳児クラス、くじら組が 4 歳児クラス、いるか組が 5 歳児クラスとなっているため、クラスごとに法律に当てはめて必要な保育士の人数を計算している。しかし時間帯によっては園児の人数が少ない場合があり、そのときは複数のクラスを 1 つのクラスにまとめることがある。例を挙げると、0 歳児が 1 人と 1 歳児が 2 人いた場合はこの 2 クラスをまとめて 1 人の保育士が担当する、ということになる。クラスをまとめた場合の担当保育士の人数は、クラスごとの担当保育士の人数を表示させる欄とは別の欄に表示させ、クラスごとの担当保育士の人数は空白を表示させる。そして空白になった担当保育士の欄は緑色で塗りつぶし、クラスをまとめていることが一目でわかるようにした。

(文責: 佐藤恭祐)

8.5 データベース

つくポスでは「おたよりはこちら」と「年賀状はこちら」を選択する画面が存在する。選択肢により読み込む住所録となるデータが変動する。つくシフトの卒園児シートをデータベースとしてつくポスへ参照する。ユーザの手順として、つくシフトの卒園ボタンをクリックすることによって、卒園児シートへ園児の情報が移動するのだがそれと同時に CSV ファイルとしても保存される。つくポスは Excel の Open メソッドにならい var workBook = excel.Workbooks.Open() によって CSV ファイルを読み込みテーブルに出力することが出来る。しかし、Microsoft の拡張機能である ActiveXObject を使用するため Internet Explorer でしか起動しないという難点もある。ただし、先方のブラウザは Internet Explorer であると確認済みのため問題なしと判断した。

(文責: 神秀樹)

第 9 章 今後の課題と展望

9.1 つくシフト第4バージョンの納品

つくシフト第4バージョンは、第3開発の納品で得られたレビューを基に改善し、1月中に納品する予定である。第4バージョンを最後の納品とする。

(文責: 佐藤恭祐)

9.2 つくポスの納品

つくポスの納品は2017年1月末に行う予定である。

(文責: 黒瀬丈哉)

9.3 つくしの子保育園での運用

我々が作成した「お助け！つくシフト」と「つくポス」はつくしの子保育園での運用を想定しており、1月に最終リリースを行う予定である。それ以降の運用についてはグループリーダーの連絡先を先方に伝えているので、問題が発生した時には連絡をいただき、その都度修正する。

(文責: 大谷笙互)

第 10 章 評価

10.1 活動内容に関する評価

システムの納期を守ることについては、何度か変更することがあり、先方に迷惑をかけてしまうことがあった。タスクを均等に振り分けることについては、何度か偏りが生じてしまったが、前期で開発を中心に担当していた人がポスター作成を担当するなど、色々な経験を積むことはできた。発表については、ポスターセッションが多かったので、何度かのポスターを作成したが、発表技術とともに、高い評価を得ることができた。

(文責: 大谷笙互)

10.2 開発内容に関する評価

10.2.1 つくシフトに対する評価

成果発表会でのアンケートで得られた、つくシフトについての評価を以下に記述する。

機能についての評価

- 以前の形式にのっとってシフト表を出すというのは、良いと思った
- 一瞬でシフトが作成されて便利
- ガントチャート機能が有り、問題が改善されているユーザインタフェースについての評価
- 手軽でわかりやすくシフト表を作れている
- ユーザインタフェースの改善があればもっと良い
- 誰もかれもが使いこなせるシステムなのか
- フィードバックに対する対応もできており、素晴らしいと思います
- 保育士さんの作業の効率化に役立っていると思いました
- システムとしては問題を解決できていると思う
- 保育園だけでなく、学校や他の施設でも利用できそうだと思った
- ユーザーの要望を良く理解し、製作しているので、実に実用的なソフトができており、好感が持てる

(文責: 佐藤恭祐)

10.2.2 つくポスに対する評価

成果発表会でのアンケートで得られた、つくポスについての評価を以下に記述する。

- つくポスの操作が難しそうに感じた
- 他にリリースされているソフトとの違いがわからなかった

(文責: 黒瀬丈哉)

10.3 評価のまとめ

評価のまとめとして、「つくシフト」についての評価は「ユーザーの要望を良く理解し、製作しているので、実に実用的なソフトができており、好感が持てる」「保育士さんの作業の効率化に役立っている」「一瞬でシフトが作成されて便利」など、高い評価を得ることができた。一方で、「ユーザインタフェースの改善があればもっと良い」「誰もかれもが使いこなせるシステムなのか」など、解決していくべき問題点も挙げられた。「つくポス」についての評価は「他にリリースされているソフトとの違いがわからなかった」「つくポスの操作が難しそうに感じた」「もっとユーザインタフェースを改善できると思う」など、これから解決していかなければならない課題が浮き彫りとなる形となつた。

(文責: 渡邊隆広)

第 11 章 学び

11.1 タスク管理

本グループでは、タスク管理に Redmine を使用していたが、前期活動では更新をしないことが多く、効果的に活用できていなかった。そこで、後期からはプロジェクトの活動の始めと終わりに全員が Redmine を更新する時間を設けた。また、定期的にグループリーダーが整理するようにした。その結果、タスク管理をより効果的に行うことができた。

(文責: 大谷笙互)

11.2 情報の洗い出し

私たちは 5 月 16 日につくしの子保育園で第 1 回保育訪問を行った。その訪問から得られた情報をもとに第 1 開発に取り組んだが、この訪問で得た情報が不足していたため、開発に支障をきたした。私たちが開発したつくシフトには保育士情報登録画面があり、その画面で勤務シフト表を作成する際に必要な保育士情報を登録するのだが、保育士のどのような情報が必要かを先方から聞き出せていなかった。そのため保育士情報登録画面の開発が一時止まってしまい、次の開発に進むことができなくなってしまった。このことから私たちは、事前に作成するシステムの構成を明確にしてメンバーで共有し、システムを完成させるために必要な情報を漏れがないようにメンバー全員で洗い出すことの重要性を学んだ。6 月 21 日に行った第 2 回保育訪問ではこの学びを生かし、事前にメンバーで先方から聞き出すべき情報をしっかりと洗い出してから訪問に臨んだ結果、必要な情報を洗い出すことができ、その後の開発を問題なく進めることができた。

(文責: 神秀樹)

11.3 先方とのコミュニケーション

我々が今回フィールドに設定したつくしの子保育園にいる職員さんは、IT に詳しくない方であるので、我々が常識的に使っている言葉でもうまく伝わっていないことがあった。例えば、エクセルでよく使われている「セル」や「シート」といった言葉が伝わらないということがあった。そこで実際にパソコンでその場所を示しながら説明した。このことから私たちは先方とコミュニケーションをとる際、言葉だけでなく、補足で説明できるような資料やシステムのプロトタイプを用いて説明することがコミュニケーションにおいて重要なことであると学んだ。

(文責: 大谷笙互)

11.4 スケジュール管理

本グループの当初のスケジュールでは前期につくシフトについて取り組み、後期につくシフトについて取り組むことになっていた。その際、前期に取り組んだシフト作成支援システムの要求分析が不十分であったことや、レビューをいただくことで新たな要求が生まれてきた。それについて考慮したスケジュールを立てることができていなかったため、後期にもシフト作成支援システムについての活動が発生してしまい、後期のタスク量が増えてしまった。このことから我々は、要求分析の重要性とともにレビュー後に新たに要求が増えることもあるので、柔軟に対応できるスケジュールを立てる必要性について学んだ。

(文責: 大谷笙互)

11.5 システムテストの重要性

私たちはこれまでにつくシフトの納品を3回行った。第1回納品の際にはそれぞれ開発した機能をマージしてから納品を行った。マージ後には一通りに動作は確認したが、テスト項目等を用意して行ったわけではなかったため、抜けが生じていた。そのため第1回納品直後に先方からバグが発生したとの連絡を頂いた。私たちはシステムのテストが不十分であったためバグの発生に気が付かなかつたと考えた。このことから私たちはつくシフトのテスト項目を作成することとした。テスト項目はシート毎に分類分けをした。テスト項目作成方法としてはGoogle ドライブ内にチーム内共用のスプレッドシートを作成することとした。これにより、いつでもチーム内の誰でもシステムのテストを行うことを可能とし、誰が行ったのかも確認できることができるようになった。第2回納品以降は以上のテスト項目を使用し、システムのテストを行うことで納品後のバグの発生が減ったことが確認できた。私たちにはつくしの子保育園というユーザーが存在し、依頼を受けている以上はそのユーザーにも使ってもらうということをしっかりと意識しなければならないという学びを得た。

(文責: 小嶋凌弥)

11.6 ポスター作成

本グループでは、成果発表会に向けて計3回ポスター作成をした。これらのポスターは教員やTAにレビューをいただきながら改善をすることで、よりよいポスターを作ることに挑戦した。ポスターを効果的に使うにはどのようにすればいいか考え、文字の量やイラストのデザイン、配置などについて学んだ。

(文責: 大谷笙互)

第 12 章 まとめ

本グループではつくしの子保育園を対象に、事務作業のうち、勤務シフトの作成と卒園児向けのはがき作成を支援するシステムを開発した。勤務シフト作成を支援するシステム、「お助け！つくシフト」では、Excel VBA を用いた。現状のシフト作成の様式を極力変えずに、職員さんでも操作に慣れやすいように Excel で操作することを想定したためである。また、「つくポス」では、はがき作成における宛名面の作成を簡単にできるようなシステムを開発した。つくシフトの園児情報を参考にすることで、名簿管理を簡単にできるようにした。また、送り先の選択を保育士からの情報を元に、簡単にできるように実装した。このプロジェクトでは、フィールドに即したシステムを作ることが目的である。先方とのコミュニケーションを通じてニーズの把握や、その分析について実践的に学ぶことができた。また、システム開発をする上で、要件定義や開発も経験することができ、多くの学びを得ることができた。本プロジェクトでの学びを今後の活動に活かしていきたい。

(文責: 大谷笙互)

付録 A 新規技術習得

- ExcelVBA
- HTML5
- JavaScript
- Redmine
- Git
- GitHub
- アジャイル開発手法の一つであるスクラム
- TeX
- Adobe Illustrator
- テスト項目 r

(文責: 渡邊隆広)

付録 B 活用した講義

- ソフトウェア設計論 I
- ソフトウェア設計論 II
- プロジェクトマネージメント

(文責: 黒瀬丈哉)

付録 C テスト項目

No	項目	試験内容	結果
1	図表登録	名前を空けて登録	未のままフォームが消失して何も記入しない
2	1_1	登録ボタン	登録ボタンを押しても登録
3	1_2	確定ボタンを空けて登録	登録ボタンを空けて登録
4	1_3	確定ボタンを空けて登録	登録ボタンを空けて登録
5	1_4	誕生日を空けて登録	誕生日を空けて登録
6	1_5	年、月を空けて登録	年、月を空けて登録
7	1_6	年、日を空けて登録	年、日を空けて登録
8	1_7	月、日を空けて登録	月、日を空けて登録
9	1_8	年のみを空けて登録	年のみを空けて登録
10	1_9	月のみを空けて登録	月のみを空けて登録
11	1_10	日のみを空けて登録	日のみを空けて登録
12	1_11	郵便番号を空けて登録	郵便番号を空けて登録
13	1_12	住所を空けて登録	住所を空けて登録
14	1_13	名前を空けて登録	名前を空けて登録
15	1_14	会員登録登録用の場所セルやボタンをクリック	会員登録登録用の場所セルやボタンをクリック
16	1_15	確認用登録用の場所セルやボタンをクリック	確認用登録用の場所セルやボタンをクリック
17	1_16	確定用登録用の場所セルやボタンをクリック	確定用登録用の場所セルやボタンをクリック
18	1_17	郵便番号登録用の場所セルやボタンをクリック	郵便番号登録用の場所セルやボタンをクリック
19	1_18	名前を空けて登録	フォームが消失
20	1_19	確認入力の値で登録	全く同じ名前の入力を検索する場合、シートの上方から同じ名前の入力を検索で検索
21	1_20	名前以外で登録(住所、郵便番号)	登録
22	1_21	確認用登録用の値で検索	確認用登録用の値で検索
23	1_22	登録用登録用の値で登録	登録用登録用の値で登録
24	1_23	確定用登録用の値で登録	確定用登録用の値で登録
25	1_24	誕生日を空けて登録	誕生日を空けて登録
26	1_25	年、月を空けて登録	年、月を空けて登録
27	1_26	年、日を空けて登録	年、日を空けて登録
28	1_27	月、日を空けて登録	月、日を空けて登録

図 4.1 作成したテスト項目

(文責: 小嶋凌弥)

参考文献

- [1] つくしの子保育園～子どもたちが毎日のびのびとすごす函館の元気な保育園です！
<http://www.tukusinoko.com/>
- [2] 保育分野における人材不足の現状 - 厚生労働省
www.mhlw.go.jp/file/06-Seisakujouhou-11600000.../0000057759.pdf
- [3] 保育士確保
<http://www.mhlw.go.jp/bunya/kodomo/osirase/140131-1.html>
- [4] 保育所等関連状況取りまとめ
www.mhlw.go.jp/stf/houdou/0000135392.HTML
- [5] 郵便番号検索 API
<http://zip.cgis.biz/>