

Googleアプリを用いた装置予約システムの構築

○丹野 健徳

東北大学 電気通信研究所 研究基盤技術センター

1. はじめに

当センターでは、走査型電子顕微鏡やX線回折装置などの実験装置の共通利用を提供している。各種装置の利用予約は、当センターの管理室前に設置した予約簿へと直接記入するか電話等で担当者に連絡することで行っていた。しかし、今般の新型コロナウイルス感染症の広がりを受けて、大学施設への立ち入りなどに様々な制限が生じた。当センターにおいても、利用者の入構制限や担当者のテレワークによる不在などで、装置の利用予約が滞った。

円滑な共通利用の提供を行い、同時に利用者-管理者間での接触を最低限に防ぐために、装置の利用予約をオンラインで行う必要が出てきた。そこで、東北大メールに紐づいて利用できるG Suiteの機能であるGoogleアプリ（ドライブ、フォーム、ドキュメント、スプレッドシート、サイト、カレンダー、メール）とアプリと連携して利用できるプログラミング言語であるGoogle Apps Script（以下GASと表記）を用いて、装置の利用予約システムのオンライン化を行ったので、これについて報告する。

2. システムの概要

システムの構築にあたり、最初に利用予約システムに必要な動作を想定することから始めた。次に、それぞれの動作についてGASで構築していき、最後に全てのGASを連動して動作させるようにして利用予約システムを構築した。想定した動作は以下の通りである。

1. フォームに入力した内容がスプレッドシートに保存
2. 各種の利用装置毎にプログラムを分岐
3. スプレッドシートの内容をカレンダーに反映して予約完了
4. 予約完了時、または予定が重複した予約失敗時に通知メールを送信
5. 予約内容の控えを当センターと利用者の所属研究室の責任者に送信

それぞれの動作は以下のフロー図に従って処理される。これらをGASの内容の一部について以降の項目で説明していく。

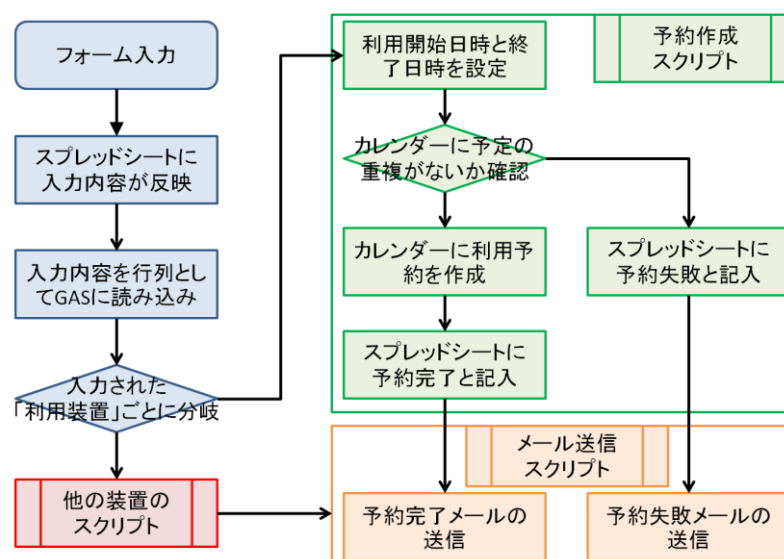


図1 装置利用予約システムのフロー

3. システムの構築

3.1 フォームとスプレッドシート

フォームに入力した内容は、自動的に関連付けたスプレッドシートに内容が記録されていく。さらに、そのスプレッドシートと GAS を紐づけることで、入力された内容をプログラムの引数として利用しやすくなっている。

各種の利用装置毎にプログラムを分岐させる関係上、プルダウンからあらかじめ入力してある利用装置名を選択させることで、入力による表記ゆれをなくして分岐条件の引数として利用しやすくした。また、所属研究室の入力もプルダウンとしている。利用申請を受けた研究室のみを選択肢とすることで、他研究室の利用制限、利用者の管理を容易にするためである。

スプレッドシートの入力内容は、1 行 8 列の行列 **values** として GAS 上で宣言することで、行列内に保存された各入力内容 **values[0][i]** を引数として利用できる。スプレッドシートの入力内容と行列番号、GAS 上での主な使用用途について表 1 にまとめる。

表 1 スプレッドシートの入力内容と行列番号

スプレッドシートの入力内容	行列番号	GAS 上での使用用途
タイムスタンプ	values[0][0]	フォームの入力時間
メールアドレス	values [0][1]	メールの送信先
利用者名	values [0][2]	ドキュメントで置換
所属研究室	values [0][3]	CC でのメールの送信先
利用装置	values [0][4]	装置毎にプログラムの分岐
利用予定日	values [0][5]	カレンダーでの 重複の確認と予定の作成
開始時間	values [0][6]	
終了時間	values [0][7]	

3.2 利用装置毎のプログラムの分岐

プログラムの分岐は、if 文と else if 文を利用することで実現可能である。今回作成した GAS の場合では、利用装置毎にプログラムを分岐させるので、入力された **values[0][4]** の内容が if 文内の特定の文字列と一致すれば、予約作成プログラムに進む。一致しない場合は、次の else if 文に進み、他の装置の予約作成プログラムに進むように構築した。

利用者が予約フォームから入力した際に、利用装置名を間違えた場合や、表記ゆれがある場合はプログラムの分岐が上手くいかなくなる。そのため、利用装置の入力は、プルダウンメニューから行うこととした。

3.3 予約内容をカレンダーに反映

前段階の準備として、自身の東北大メールの Google カレンダーアプリに、各装置について個別にカレンダーを作成しておく必要がある。次に、各カレンダーの『設定と共有』から、カレンダーIDを確認する必要がある。GAS 上でカレンダーIDを宣言することで、カレンダーに GAS からアクセスして予定の作成などが可能となる。

カレンダーに予定を作成する上で、予定の開始日時と終了日時を設定する必要がある。今回作成した GAS の場合では、スプレッドシートに入力された **values[0][5]**, **values[0][6]**, **values[0][7]** が、それぞれ利用予定日、開始時間、終了時間にあたる。日時の変数は通常の変数とは参照の仕方が異なるため、**new Date** で新たに生成する。また、生成された日付の変数は標準では 00 時 00 分 00 秒となっているので、**setHours** 等を用いて時間を入力していき、開始時間と終了時間について YYYY 年 MM 月 DD 日 hh 時 m m 分という内容を設定する。

次に、設定した開始時間と終了時間との間でカレンダーに予定の重複がないかを確認して、重複がなければ利用予約の内容をカレンダーに予定として作成する。予定の作成が出来たならば、‘予約完了’とス

ブレットシートセルに入力して、後で確認が取れるようにした。また、予定の重複があった場合は、‘予約失敗’と入力するように構築した。これらの入力項目は、次項の予約完了時と予約失敗時のメール送信の分岐とし使用する。

3.4 予約完了時、予約失敗時の通知メール送信

予約内容 GAS にはメール送信の方法が 2 種類用意されており、MailApp.sendEmail と GmailApp.sendEmail という 2 種類のアプリケーションである。GmailApp.sendEmail の方が送信元の設定などの詳細な設定が可能である。例えば、GmailApp.sendEmail のオプション欄である{ }内に、bcc を追記することで、予約内容の控えメールを自身のアドレスなどに送る設定が出来る。さらに、送信元のメールアドレスは、デフォルトでは自身の東北大メールアドレスとなるが、オプション欄に noReply:true を追加することで、noreply@tohoku.ac.jp が送信元となる。さらに、Gmail アプリの設定で、他のメールアドレスを追加することが可能で、送信元を追加したメールアドレスに設定できる。以下にプログラムの例文を示す。

```
GmailApp.sendEmail(user@tohoku.ac.jp, Title, Body, {bcc:'XXX@tohoku.ac.jp', noReply:true});
```

予約完了時と予約失敗時のプログラムの分岐は、3.3 でスプレッドシートセルに入力した‘予約完了’と‘予約失敗’を if 文 else if 文の条件とすることで分岐させた。また、メール本文の内容は、ドキュメントで定型文を用意しておき、その中の一部をフォームで入力された内容に置換することにした。カレンダーの場合と同様にドキュメントを GAS で利用する場合も、ドキュメント ID を確認して GAS 上でドキュメント ID を宣言する必要がある。図 2 に Google ドキュメントによるメール本文と送信されたメール文を示す。ドキュメント内で { } で囲まれた部分が置換されていることがわかる。

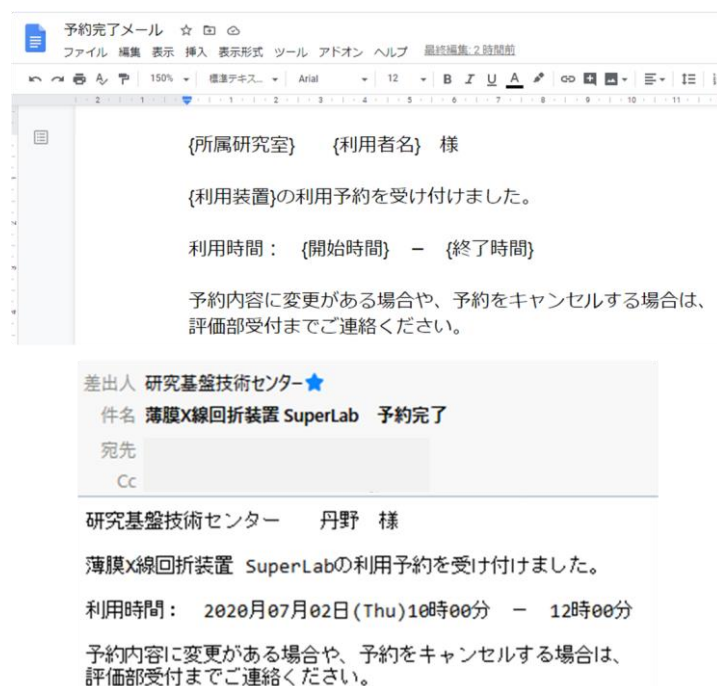


図 2 ドキュメントによるメール文と実際の送信メール

4. システムの運用

装置予約システムの運用を行うために、Google サイトにカレンダーとフォームを埋め込んで、ホームページを公開した。Google サイトを用いることで、カレンダーとフォームの埋め込みが簡便に出来る。それに加えて、一般の方にホームページが表示されずに、東北大メールアドレスをもった教職員と学生

のみが閲覧できるように設定できる。図 3 に実際に運用している装置予約システムの画像を示す。

当初は利用申請を受けた研究室のみが閲覧や予約ができるようにすることを検討したが、共有化の設定やプログラムが煩雑になるため実装はしなかった。その代わりに、所属研究室の入力欄を、利用申請を受けた研究室のみが表示されるプルダウンメニューとすることで、それ以外の研究室が利用予約を出来ないようにしている。

所属研究室ごとに各研究室の利用責任者のメールアドレスをデータベース化しており、所属研究室を入力することで、研究室の利用責任者にも CC でメールが送信して、学生が実験していることを研究室側でも把握できるようにした。

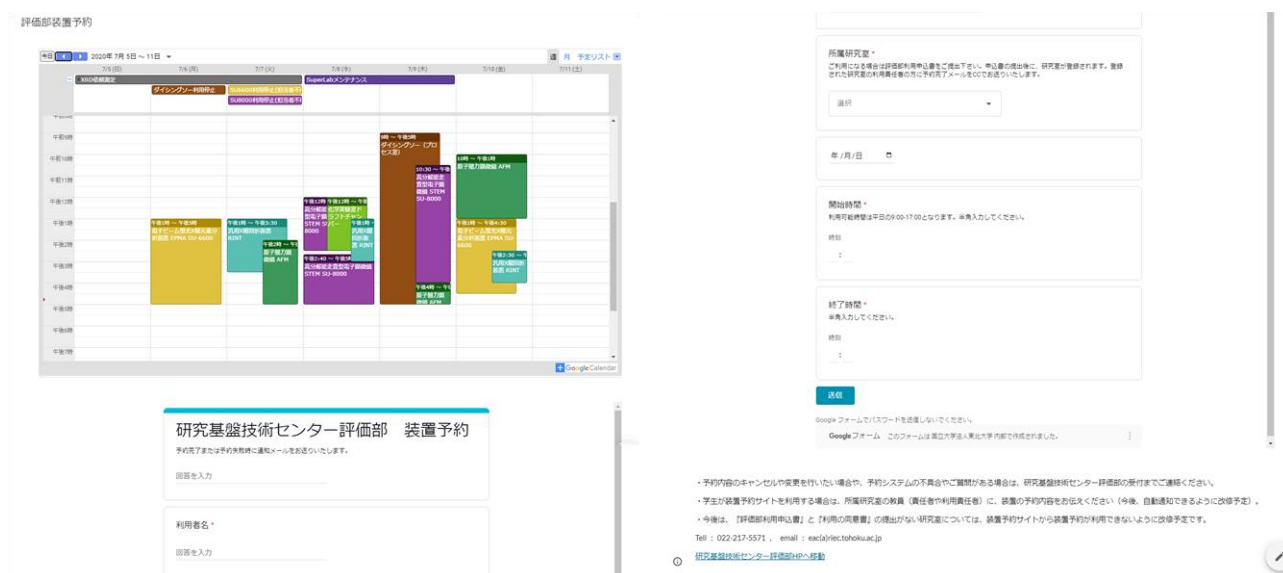


図 3 実際に運用している装置予約システムの画像

5. まとめ

私自身のこれまでのプログラミングの経験は、C 言語を用いた簡単なプログラミング実習を学生時代に受講したことがあり、測定結果の解析にテキストユーザーインターフェースのソフトを利用していること程度である。プログラミングの知識は多少有しているが、本格的なプログラミングは行ったことはなかった。今回、利用予約システムのオンライン化にあたり、参考文献^[1]と GAS のホームページ^[2]もとに GAS の構築を行った。本格的なプログラミングの経験がなくても、各種 Google アプリを連携させることで、はじめからプログラミングを行うよりも簡単にシステムを構築できた。

さらに、管理者側から装置のメンテナンス予定等を入力するシステムも構築した。今後も適宜、機能の更新や、他の GAS の利用方法などを応用していきたい。

参考文献

- [1] 高橋宣成 (2018)『詳解!Google Apps Script 完全入門』秀和システム
- [2] Google Apps Script: <https://developers.google.com/apps-script>