

# LA 決定アシストシステム

牧野 浩之

## 1 はじめに

知的システムデザイン研究室には LA という日直の制度がある。LA は研究室生活において発生する雑務を共同して負担できるように設けられている。現在、LA の決定は紙ベースで希望調査や集計が行われている。その仕組みは、メンバーそれぞれが LA になることができる日を一覧表に記入し、LA 担当が公平に割り当てられるように集計を行うというものである。この作業は月に 2 回程度行われており、ルーティンワークとなっている。本システムでは、この LA の決定作業に着目し、シフト表のペーパーレス化や煩雑な作業における負担が軽減できるサービスを提供する。

## 2 LA 決定アシストシステム

### 2.1 システム要件

LA を決定する際に LA 担当はメンバーのスケジュール希望に応じてどの日に誰を割り当てるか、どのような頻度で割り当てるかを考えなければならない。また、決定後、メンバーの都合によってはシフトの交代を行い、シフト表を更新する作業が発生する。本システムでは以上で述べた作業をコンピュータが行い、LA 担当は最低限の確認作業で済むようにする必要がある。

### 2.2 機能要件

本システムを実現するうえで以下のような機能が必要になる。

- メンバーがスケジュールを入力できる機構
- LA 担当がメンバーの希望を採り入れながら集計を行う機構
- 交代の要望が出た際に交代の手続きを行い、シフト表を更新する作業

### 2.3 システムの概要

本システムでは、フロントエンドのインタフェースにウェブベースの Java Servlet エンジンを利用している。このため、ユーザは普段利用しているウェブブラウザで利用することができる。Servlet エンジンには、Apache Tomcat を利用している。

また、ユーザのデータはすべてサーバ側のデータベースに保持されている。本システムではデータベースサーバに MySQL を採用した。Fig. 1 に示すように、Servlet

エンジンとデータベースサーバは JDBC ドライバを用いて連携している。

そして、ユーザが利用時に、ブラウザから Servlet に GET や POST などの HTTP リクエストを送ることでよりサーバ側が処理を行うようになっている。

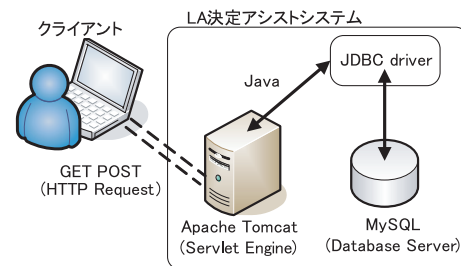


Fig. 1 システムの構成 (出典：自作)

### 2.4 利用方法

このシステムの利用方法を以下に示す。なお、ユーザの立場は、スケジュールを登録してシフトを閲覧するメンバー側と、集計を行い、LA の決定を行う集計担当者側の 2 つに分けられる。ログイン画面は共通しており、ログインを行うとメニューが現れる。各ユーザはメニューから目的別の機能を利用する。スケジュールの入力は、カレンダーの日付をクリックすることにより、その日における LA 担当の可否が登録できる。なお、集計が終わっている日、つまり、LA がすでに決定している日の予定は変更ができない仕組みになっている。

- メンバー側
  - 氏名を選択、パスワードを入力しログインを行う
  - 自分のスケジュールを入力する
  - シフト決定後、スケジュールを確認する
- 集計担当者側
  - 氏名を選択、パスワードを入力しログインを行う
  - メンバーがスケジュールを入力できていることを確認する
  - 集計期間を選択し、集計を行う
  - メンバーの都合が見つからない場合に、要望に応じて交代の処理を行う

## 2.5 集計処理の流れ

集計を行う際の段取りは以下のようにする。

まず始めに、あらかじめメンバーにどの期間からどの期間までのスケジュールを入力しなければならないかを伝えておき、スケジュールを登録してもらう。集計担当者は一覧表示でメンバーのスケジュール入力完了しているかを確認できる。なお、スケジュール未入力のユーザは自動的に×となるが、スケジュール入力画面への最終アクセス日時も同時に表示させることにより、入力し忘れを識別できるようにしている。

次に、メンバーのスケジュール入力状況を確認後、集計の開始日と終了日を選択する。なお、開始日はLAのシフトが決まっていない期間の始めが自動的に設定される仕組みになっている。集計期間を確認し、集計ボタンをクリックすると集計が行われる。集計のアルゴリズムについては2.6節で詳述する。集計後、シフト表示画面にて結果を確認できる。また、メンバー全員の都合がつかず集計システムで決定できなかった日に関してはLA担当が手動で割当を行うことができるようになっている。

## 2.6 LA 決定のアルゴリズム

LA が公平に割り当てられるようにそれぞれのメンバーにカウンタ(重み)を付けて計算している。基本的に、水曜日は全体ゼミがあり、議事録などの仕事が増えるため、1.5を加算し、それ以外の曜日は1.0を加算する計算を行っている。

決定のアルゴリズムは、それぞれの日付において、LAになることができるメンバーのうち、この重みが小さな者からランダムに2人を選ぶようにしている。選出するとすぐにユーザデータベースの重みのカウンタが加算され値が大きくなるため、日が連続して選出される可能性は低くなっている。

## 2.7 交代処理

実際の運用においては、メンバーの都合がつかなくなりLAを交代してもらう、という状況が発生する可能性がある。その際は、LA担当のメニューに現れる「LA交代」のボタンをクリックすることにより、交代することができる。操作方法は、交代したい人同士の担当している日付を選択し、氏名を選択する。次に、確認画面で承認を行うと、シフトデータベースのデータが書き換わるという仕組みである。

## 2.8 拡張機能

現時点でのリリースバージョンには含まれていないが、本システムの拡張機能として、以下のような例が考えられる。

- メール通知機能 ...担当の日が決定すると各メンバーにメールで通知される機能

- スケジュール登録機能 ...各メンバーのサイボウズの予定に担当の日を自動的に書き込む機能

## 2.9 実行画面

実行画面のイメージは次のようになっている。Fig. 2に示すイメージは、ログイン後のメニュー画面である。ユーザはここから目的のボタンを選択し、操作を行う。ユーザデータベースに管理者フラグの付いているLA担当のメニューは、集計機能と交代機能が追加されて現れるようになっている。



Fig. 2 メニュー画面 (出典：自作)

Fig. 3に示すイメージは、スケジュール入力画面である。画面はカレンダー形式になっており、該当の日付のボタンをクリックすることにより、LAになることができる日の希望を入力することができるようになっている。



Fig. 3 スケジュール入力画面 (出典：自作)

## 3 まとめ

本報告では、Java Servlet を用いて、LAの決定をアシストできるシステムを構築した。これにより、LA担当の負担が軽減できるほか、ユーザ側にとってモリアルタイムにウェブからシフトの確認やスケジュールの入力が可能になるため利便性が向上すると考えられる。また、メール通知機能など紙ベースでは成し得なかった付加価値を提供することも可能となる。今後の課題として、ユーザインタフェースの改良や他のシステムと協調して連携可能にするといったことが考えられる。