ウェブアプリケーションを用いた大学での LifeHack の検討

06H112 湯川正洋

目次

1 Introduction 1

2 Framework 2

3 実験概要 3
- 3.1 40:10 分時間割 . . . 4
- 3.2 朝会 . . . 5
- 3.3 daily 記録 . . . 6

4 実験 7
- 4.1 時間割 . . . 8

5 LifeHack 用ウェブアプリケーションの開発 9
- 5.1 開発の目的 . . . 9
- 5.2 ウェブアプリケーションによるメリット . . . 10
- 5.3 各機能 . . . 11
- 5.4 開発環境 . . . 16

6 Result and discussion 17

7 Conclusion 17

付録 A 付録 1 17
- A.1 付録 1.1 . . . 17

概要

ここに論文の概要を書く、typeset 後に A4 の 1/3 程度が目安。

1 Introduction

仕事場や私生活における生産性を向上させる方法が、‘LifeHack’ や ‘仕事術’ など (以後本論文ではこれらの方法を ‘LifeHack’ で統一する) と呼ばれ近年人気を博している。LifeHack の範囲は多肢に渡り、以下にいくつかの例を挙げる。

- アプリケーションやシェルスクリプトによるコンピュータでの自動処理
- 情報化社会において必要な情報を効率よく得るための整理
- 手帳や Web アプリケーションを用いたスケジュール管理
- 起床時間や体重を記録していくことによる私生活の改善
- 睡眠や運動による身体機能や脳機能の改善

このように LifeHack では、文字通り生活全般の改善方法について広く取り扱っている。実践される場所もまた自宅、仕事場、通勤路などさまざまである。

しかし大学においての LifeHack の例[*1]はあまり見ない。私はその理由を情報の不足と実践の制限との 2 つだと考えている。LifeHack という分野は主にビジネス誌や社会人向けウェブサイトで取り扱われている。そのため学生向けの情報があまり発信されておらず認知度が低い。また学校は仕事場や自宅に比べて授業時間や個人のスペースの狭さなどの制限が多い。そのため個人で改善できる部分が少ないか、狭い範囲に限られてしまうのではないだろうか。とは言え、LifeHack は今や多くの場で実践されており、大学でも実践する価値はあるはずである。

そこで LifeHack を大学内の私が所属する研究室に導入することで、学生の生産性にどのような影響があるのか実験を行った。またその実験によって得られたデータを元に、大学向けのより効果的な LifeHack を検討する。

[*1] 大学に限らない LifeHack はのぞく

2 Framework

実験を行うにあたって、大学向けに応用した LifeHack を導入する必要がある。最大の効果を得るには研究室の内情に即したものが最も良い。しかし研究室は研究室ごとの独自色が強い。研究室の内情に個別に対応していくと、'大学での実践例' から大きくはずれる可能性がある。研究室によりすぎず、あくまでも大学生活一般において有用なものを導入すべきである。

そこで、学生 1 人 1 人に対応させるのではなく、研究室に所属する学生全体を 1 つのチームととらえることにした。チームでの生産性を上げる LifeHack を応用することで、研究室に所属する学生全体の生産性を上げることができるだろう。

LifeHack を応用するにあたって、以下の要素を特に重要視した。

(a) 記録
(b) チームでのスケジュール管理
(c) ウェブアプリケーションを用いた LifeHack の効率化

(a) 記録

その日 1 日の自分の行動や、成果などを記録する。記録することで、自分がどの作業に時間を取られたか、無駄な時間を把握することができる。

記録という手法は世間で幅広く使われており、自己の管理や成長に効果的である。スポーツではタイムやスコアを記録し、それを参考に訓練することはよく知られている。またスポーツに限らず、ダイエットや禁煙においても効果的であることが認められている。

(b) チームでの 1 日のスケジュール管理

チーム全体で、メンバーの 1 日のスケジュールを把握できる環境作りを行う。チーム内でお互いのスケジュールやその中の作業を把握することで、お互いの作業を可視化できる。可視化することで、お互いの作業に対する助言や提言を行いやすくなる。チーム内でお互いに助言・提言を行うことで、自分だけでは気づかなかった点や無駄に気づくやすくなり、作業の効率化を行うことができる。

(c) ウェブアプリケーションを用いた LifeHack の効率化

(a) と (b) をウェブアプリケーションを用いて効率化を図る。コンピュータにより一部を自動的に処理することにより、LifeHack を行う上での負担を軽減することができる。

具体的にはチーム内の 1 日のスケジュールをまとめ、同時に記録するウェブアプリケーションを開発する。チーム全員分のスケジュールを把握することは、非常に手間と時間がかかってしまう。そこでコンピュータを用いて、できるだけ人の手がかかる部分を排除する。

3　実験概要

大学内の研究室を対象として LifeHack を実践するにあたり、以下の 2 つの条件を考慮しなければならない。これらの条件は今回実験を行う大阪産業大学工学部情報システム工学科を基準としているが、他の一般的な 4 年生大学と大差はない。

- 大学の時間割
- 学生ごとの履修状況の違い

大学の授業は時間割で区切られ行われている。そのため学生が大学でのスケジュールを組むときは、基本的に時間割ごとに時間を区切って行う場合が多い。授業が連続で 1 日中あるわけではなく、授業の間に自由な時間が取れることもある。

また学生ごとの履修状況も異なり、人によって授業がある時間とない時間が存在する。このためある時間は授業、ある時間は自由、というように単純に仕切れるものではない。

以上を踏まえて、前項で述べた要素を満たすために、今回以下の手法を用いることにした。次項より詳しい説明を行う。

- 40:10 時間割
- 朝会
- daily 記録

3.1　40:10 分時間割

この手法は 48:12 分仕事術という手法を参考に、大学の時間割向けにに応用したものである。導入によってやる気・モチベーションの向上・維持を目指し、またわかりやすいスケジュール管理の目安とする。

48:12 分仕事術という手法は 1 時間を 48 分と 12 分に分け、48 分を集中して仕事をする時間 (仕事時間)、12 分を休憩時間にあてるというものである。48 分間はあらゆる雑音や邪魔を排除して、仕事をひたすら片付ける。12 分間は 48 分間に取った行動とは違う行動を主とした休憩を取ることで、やる気・モチベーションの維持にあてる。

重要なのは 12 分間の休憩であり、必ず取らなければいけないことである。また 48 分間とは違う行動というのも重要である。例えば 48 分間デスクワークを行ったならば、12 分間はデスクから離れて体を動かす必要がある。そうすることで脳と身体を活性化し、モチベーションのダウンを防ぐ。

48:12 分仕事術は 1 時間という区切りで行われているが、大学では時間割を考慮する必要がある。私が所属し、今回実験対象となっている大阪産業大学の授業は 1 時限につき 90 分間であり、間に 10 分間の休憩がある。そこで、授業の 90 分と休憩の 10 分を足して 100 分とし、それを 2 つ分け 50 分が 2 つという風に考える。その 50 分に 48:12(4:1) で分けることで、40 分間の仕事時間と 10 分間の休憩時間にあてることにした。これにより大学の時間割と 48:12 分仕事術の共存が可能となる。

1 コマを 40 分 (+10 分) とし、2 コマで 1 時限分とする。また便宜上 1a・1b のような名称にし、対象時限数に a と b を付加することで、その 1 時限分の前半と後半とする。

1 日のスケジュールを立てる際には、この 1 コマを単位とし、1 コマごとの作業予定をあてるようにする。チーム全体でスケジュールを把握するにあたって、共通の単位を用いたスケジュールを立てることで、よりわかりやすくなる。

3.2　朝会

大学の 1 限目が始まる時間は 0900JST であるが、その 30 分前の 0830JST からチーム全員での会議 (以下朝会) を 20 分間で行う。

朝会では毎日ランダムで議長を選出し、議長は朝会の司会とまとめを行う。ランダムで議長を選出することにより、メンバー全員がほぼ等しく議長になる可能性を得る。毎日ランダムで選ばれるということから、メンバー全員に緊張感と責任感を持たせることができる。また時間が限られているため、議長がいかに朝会を進めるかという点も重要で、時間を意識したファシリテーション技術が養われる。

またメンバーは朝会が始まるまでに、スケジュールを立てておく必要がある。スケジュールは 40:10 分時間割で設けた新たな時間割を用いる。

朝会で行うことは主に以下の 2 つである。

- 個人の主張
- スケジュールの把握

個人の主張部分では、個人がチーム内のメンバー全員に向かって自由な発言を行う。内容は問題提起から昨日あったイベントの話などさまざまである。時間がかかりそうな場合や、会議が必要な場合は、別に時間を取って続きを行う。時間を取る場合は緊急をのぞいて、別の日がのぞましく、事前連絡をした方が良い。当日の場合、は他人の既に立てられているスケジュールに影響を与えてしまうからだ。

スケジュールの把握部分では、議長が決めた順番に従ってその日 1 日のスケジュールを発表し、発表者以外のメンバーは助言・提言・質問を行う。こうすることで、自分の立てたスケジュールを客観的に見てもらうことができる。客観的に見てもらうことで、スケジュールの無駄の指摘や作業の助言・提言をしてもらえる可能性がある。にさまざまな知識や技術をもったメンバーが混在するチームで、公に他人の意見を聞ける場を設けることで、他人の知識・技術を有効利用することができる。

朝会が終わり次第、議長はチームで共有している ML(メーリングリスト) に各自のスケジュールと個人の主張のまとめをメールで送信する。これによってチーム内で他のメンバーのスケジュールと主張を共有することができ、またメールという形を用いているため整理や参照が容易である。

3.3 daily 記録

40:10 分時間割で新たに設けた時間割を用いて、帰宅前に毎日の行った作業を記録する。自分がスケジュール通り作業を行うことができたかどうか確認できる。また毎日の記録を集計していくことで、実際に自分がどの程度のペースで作業を進められているかの把握が容易となる。

この daily 記録は日報という形でチーム内のメンバー全員に共有される。この時 40:10 分時間割の作業記録とは別に、その日の成果や反省、今後の目標を記録し共有する。メーリングリストによるメールという形で行われるので、各メンバーが別々の場所にいても携帯電話やコンピュータで把握することができる。

記録を取る際にはできるだけ具体的な方が良い。例えば 1 コマで読んだ本の文字数や、走ることの出来た距離などである。具体的であればあるほど今後の参考にしやすく、自己の成長に役立てることができる。またコマ単位で管理することで、時間帯や体調によってどれだけ影響を受けるかがわかりやすくなり、自分のことを知りやすくなる。

4　実験

実験は私の所属する情報安全工学研究室内で以下の条件で行なった。期間はグレゴリオ歴 2009 年 8 月 1 日から 10 月 31 日までの 3 ヶ月間である。

- 参加した学生は私を含め 10 名
- 参加学生は全て 4 回生で情報安全工学研究室所属
- 履修状況に違いがある (授業のある学生とない学生がいる)

1 日の主なスケジュールとしては以下の図の通りである。

#ここで図が入る

0830JST に始まる朝会までに参加者は 1 日のスケジュールを立てておく。その後朝会を行ない、議長の選出、個人の主張、スケジュールの把握を行なった後 0850JST までには終える。0900JST 以降は研究室に合わせた 40:10 分時間割 (詳しい時間割は次項参照) にのっとってスケジュール通り行動し、帰宅前には daily 記録を行なった上で日報として ML にメールを送信する。これを平日の月曜日から金曜日まで毎日繰り返す。

4.1 時間割

40:10 分時間割を研究室に合わせ、以下の時間割のようにする。時間は全て JST である。

適度な休憩を取るという観点と、リズムを作るという観点から、3 コマの仕事時間を 1 つのセクションとし、各セクションの間には 1 時間以上の休憩時間を用意した。

朝会	0830 – 0850
1a 限	0900 – 0940
1b 限	0950 – 1030
2a 限	1040 – 1120
昼休憩	1120 – 1250
3a 限	1250 – 1330
3b 限	1340 – 1420
4a 限	1430 – 1510
自由時間	1510 – 1610
5a 限	1610 – 1650
5b 限	1700 – 1740
6a 限	1750 – 1830
自由時間	2000 – 2150

5　LifeHack 用ウェブアプリケーションの開発

今回の LifeHack の一環として、各 LifeHack を効率化するためのウェブアプリケーションを開発した。実験と同時進行で開発とアップデートを行なったため、こちらは 3 ヶ月間全体での運用にはならなかった。しかし基礎となるシステム自体は実験 2 ヶ月目には完成しており、その後細かい点でアップデートを行なったものの、概ね 2 ヶ月間運用することができた。

また今回は実験ということもあり、研究室内のコンピュータからのアクセスしか受け付けないようにした。一部ユーザー認証が必要であるが、基本的に研究室からでは誰でも使用することができる。

5.1　開発の目的

主な目的はスケジュール把握と daily 記録部分の一部自動化による、作業の効率化である。

当初朝会後にチーム全体のスケジュールを共有する際に、ML にメールで送信するという方法を取っていた。しかし問題があり、議長が一括してスケジュールをまとめたメールを作成し、送信するのでは議長 1 人にかなりの負担を与えてしまう。ただでさえ毎回ランダムで選ぶようにしているので、議長になった場合 1a 限のスケジュールに影響を与えかねなかった。また参加者全員が 1 人 1 人自分のスケジュールをのせたメールを作成し、送信するのではメールを開く手間がかかってしまう。

そこでその手間がかかってしまう部分をアプリケーションで一部自動化することにした。具体的には、参加者各人がアプリケーション側に 1 日のスケジュールを登録すると、アプリケーション側が自動で全員分をまとめる。議長はそのまとめ部分を引用し、朝会で変更のあった部分を修正して ML にメールで送信するだけにした。

同時にこのスケジュール登録機能は daily 記録に応用することができる。帰宅前にスケジュール部分を実際の行動に置き換えて登録することで、自動的にコマごとに作業項目をまとめ集計する。集計することで自分が今までにどの作業を多く行なったかや、無駄を確認でき、今後のスケジュールに活かすことができる。

5.2　ウェブアプリケーションによるメリット

デスクトップアプリケーションではなくウェブアプリケーションを開発した利点は以下の 2 点である。

- インターネット環境とウェブブラウザを持つ端末があればどこからでも使用可能
- バグ修正やアップデートが容易

チーム全体のスケジュールを改めて把握しようとする際に、特定のコンピュータのデスクトップアプリケーションを起動しなくてはいけないとなると大変不便である。しかしウェブアプリケーションの場合はインストールが不要で、インターネット環境とウェブブラウザを持つ端末からならばどこからでも使用可能で便利である。

また参加者からの要望があれば、実験と同時進行でアプリケーション自体を随時アップデートする必要がある。ウェブアプリケーションならば 1 度サーバー上のアプリケーションをアップデートすれば済む。しかしデスクトップアプリケーションの場合、アップデートが必要な度にユーザー側でアップデートする必要があり、アプリケーションをインストールしているコンピュータの数だけ手間がかかってしまう。ウェブアプリケーションとデスクトップアプリケーションを比べてみると、前者の方がアップデートが容易であると言える。

5.3　各機能

主に 4 つの機能を実装した。次項より各機能の解説を行なう。スケジュール登録の部分のみユーザー認証が必要である。

1. ユーザー認証
2. スケジュール登録
3. スケジュールまとめ
4. 記録の自動集計によるデータ集

全体の概念図は以下の通りである。
#ここで全体の概念図

5.3.1 ユーザー認証

スケジュールまとめやデータ集は誰でも見ることが可能であるが、スケジュール登録のみはユーザー認証が必要である。故意や事故による誤登録を防ぐためである。

ユーザー認証にはユーザー名とパスワードが必要である。ユーザー名は単純に参加者の苗字のアルファベットとしている。パスワードは各人自由にいつでも設定できるようにしており、認証時には MD5 というハッシュ関数を用いている。MD5 によって得られたハッシュ値と、データベースに登録されているユーザー情報のハッシュ値とで照合し、ユーザー認証を行なう。

研究室内向けの実験用ウェブアプリケーションであるので、ユーザー登録などのユーザー管理部分は今回作成しなかった。人数に変動がなく、10 名という数なので何か変更があれば、私だけで十分対応が可能だからだ。

5.3.2　スケジュール登録

40:10 分時間割に従ってスケジュールを登録する。基本的にその日の入力しか受け付けないようになっており、ここで登録されたスケジュールは、チーム全員で共有されるスケジュールまとめに登録される。1 度登録されれば終わりというわけではなく、いつでも変更や削除が可能である。

#ここでスクリーンショット

スケジュール登録と同時に、簡単な主張部分としてつぶやき機能を搭載しており、簡単なメッセージをチャットのように送信・共有することができる。

#ここでスクリーンショット

5.3.3　スケジュールまとめ

　スケジュール登録によって得られた各人のスケジュールをまとめて表示する。全てテキストベースであり、表やグラフなどは一切表示しない。これはこのスケジュールまとめをメールにコピーして送信するからである。朝会に参加しなかった者も表示するようになっており、朝会に出席したか欠席したかわかりやすいようになっている。

　#ここでスクリーンショット

5.3.4　記録の自動集計によるデータ集

コマごとの作業内容をまとめ、割合を表示する。記録開始日から最後に記録した日までの期間[*2]も表示している。項目数のしきい値を上昇させることで、項目数の少ない項目をその他にまとめることもできる。

#ここでスクリーンショット

[*2] 実験終了後も使い続けているため、最後に記録した日は人によってまばらである

5.4 開発環境

以下の環境で開発を行なった。

- ハードウェア (IBM PC/AT 互換機)
  - HDD... ... 80GB
  - CPU... ... Int$_{e}$l® Core^{TM}2 1.66GHz[x86 Family 6 Model 14 Stepping 8]
  - ロジックボード... ... YNRC-BR
  - チップセット... ... Radeon Xpress 200M
  - メモリ... ... 1GB
- OS... ... Vine Linux 4.0 [?]
- 開発言語... ... PHP 5.2.6 [?, ?]
- データベース... ... MySQL 5.0.27 [?, ?]

6 Result and discussion

3 ヶ月間の実験を経て、概ね期待通りの結果を得ることができた。同時に欠点や想定外の副次的な効果も得ることができた。利点と欠点を以下のように大きく分けた。次項より詳しく説明する。これらの結果は主に参加者に対するアンケートによって得られたものである。

¡利点¿

計画・行動・結果によるフィードバック
作業の可視化による助け合い
見られているという意識
休憩時間の有効利用

¡欠点¿

6.1　計画・行動・結果によるフィードバック

朝会のためのスケジューリングと日報のための daily 記録により、1 日の自分の計画と行動の結果がよくわかるようになった。そのため自分がどれだけ計画を達成できたかや、達成できなかったかが一目瞭然となった。

実験前までスケジュールを組んで行動していなかった参加者の変化が特に顕著だった。スケジュールを組まず、記録も取っていない場合、1 日の自分の行動があいまいにしか把握できなくなる。つまり自分がどれだけ作業に時間をかけているかや、無駄な時間を過ごしているかを正確に把握できていない。具体的に問題点をとらえられなければ、具体的な解決策に取り掛かれるはずもなく、更にいくら問題に取り組んでも結果がわかりにくい。そこで結局、的確な解決策に乗り出せていない参加者が多かった。

しかし今回の実験で、あいまいになっている部分を参加者自ら具体的に認識することにより、改善のための具体的な努力をすることができるようになった。

6.2　作業の可視化による助け合い

朝会と日報によって他人の詳しい作業内容がわかるようになり、自然とチーム内でお互いに助け合う姿勢が見られるようになった。

実験前までは、お互いの作業内容や進捗状況をほとんど把握できていなかった。特に今回実験を行った情報安全工学研究室は、今年度の卒業研究内容が 1 人 1 人違う。基本的に 1 人で始めから最後までですることになり、内容自体も専門化していくので、お互い手が出しにくかった。結果として、お互いの卒業研究の進捗状況がわかず、そもそも積極的にわかろうともしなかった。

しかし朝会と日報によって他人の行動を詳しく把握することできるようになり、自然とお互いの進捗状況がわかるようになった。すると参加者同士でお互いのアドバイスをし合うようになり、問題解決のためにさく時間が劇的に少なくなった。また複数の参加者の知恵を出し合うことで、より良いアイデアや結果を得ることができるようになった。

6.3　休憩時間の有効利用

40:10 分時間割を導入するまでは、各自が好きな時間に休憩を取っていた。休憩時間は読書、睡眠、会話、ゲームなどに費やされていた。チームが個室の中で好きな時間に休憩を取ると、下図のような休憩時間のずれが生じる。またそのチーム内の親密度が高いほど、相手の行動や言動が気になるということから、休憩時間のずれはチーム全体の生産性低下を招いてしまう。

この問題は実験前から研究室内で発生しつつあったが、40:10 分時間割によって休憩時間が全員同じなることで、作業時間中に誰かが休憩を取るということがなくなった。

7 Conclusion

Acknowledgment

参考文献

[1] RFC792, Internet Control Message Protocol

付録 A 付録 1

A.1 付録 1.1