

学生の状況に適応した学生情報システムの設計

2018SE030 川津守央 2018SE040 松井俊樹

指導教員：沢田篤史

1 はじめに

現在、情報システムが普及し大学では、学生に対して多岐にわたるサービスが Web を介して提供されている。南山大学では PORTA, WebClass, 講義資料ダウンロードサーバが利用されている。複数のサービスが存在することで、学生が必要としている情報が複数のサイトで重複してしまふことや、探す手間があり学生の負担になっている。解決するためのソフトウェア基盤が存在しないことが課題になっている。

本研究の目的は、複数の学内 Web サービスを統一的に組み合わせたアプリケーションのソフトウェア基盤を構築することで、サービスを横断せずに利用が可能になり情報の取得を容易にさせることを目指す。また昨今スマートフォンの流通により、2019 年のインターネット利用率（個人）は 89.8 % であり、端末別のインターネット利用率はスマートフォン（63.3 %）が PC（50.4 %）を 12.9 % 上回っている調査結果 [1] がある。学内のサービスをスマートフォンで見られるスマートフォン向けアプリケーションが存在しない。スマートフォンユーザ向けに学内 Web サービスを横断せず利用できるネイティブアプリケーションのソフトウェア基盤を提案する。

アプリケーションのソフトウェア基盤の構築にはマイクロサービスアーキテクチャの技術を用いる。マイクロサービスアーキテクチャは、自律性と特殊性という特徴を持っている。これにより学生が利用する統一的なアプリケーションの開発をする際に適していると考えている。

効果として、複数のサービスを横断せずに利用できることで、毎回ログインすることが無くなる。その結果、学生の必要な情報を手間なく見つけることで学生の負担が軽減できることが挙げられる。

モノリシックで構成される NaNApp と本研究で提案するソフトウェア基盤に基づくアプリケーションを比較し、考察した。

2 大学における学生情報システムの現状と問題点

2.1 大学における学生情報サービス

インターネット利用が当たり前になり、情報システムの普及してきたことで大学では、様々な用途に向けた学生向けポータルサイトや学内 Web サービスが提供されている。10 代～20 代はインターネット接続する端末がスマートフォンであり、保有率が多いことからポータルサイトやサービスをスマートフォンで利用している割合が高い。

現在、南山大学では大学側が学生に向けて情報を発信するポータルサイト、サービスが主に PORTA, WebClass, 講義資料ダウンロードサーバの 3 つ存在している。他大学の学生に学内 Web サービスについて調査を行った結果、26 大学のうち全ての大学が 1 つ以上のサービスを利用していた。このことから、インターネットの普及により大学情報システムが多く利用されていることが分かる。

2.2 南山大学の学 Web サービスの問題点

現在、南山大学では複数のサービスが混在している。必要とする情報がどのサービスにあるのか、欲しい情報の取得に時間がかかってしまうなど、様々な弊害が生じる。また通知が来ないことで教室変更や休講補講情報を講義が始まる数分前に知る学生が存在していることも課題である。

今後は課題解決と長期利用を考えてサービスの追加、変更が容易なアプリケーションのソフトウェア基盤が求められる。

2.3 NaNApp について

2017 年に南山チャレンジプロジェクトの中で NaNApp[2] が開発された。NaNApp では、主に以下の 4 つの機能が提供され利用されていた。

- 当日の授業スケジュール確認
アプリケーションのホーム画面の 1 つで時間割管理機能で登録した授業の始業、開講教室が表示。
- 時間割管理
授業を検索、選択しアプリケーション内の時間割に登録。時間割の色分け機能を使用してユーザの時間割の色を好みにデザインすることや時間割を左右にスライドすることで Q1～Q4 までの授業確認可能。
- 休講補講情報のお知らせ
時間割に登録をすることで、休講補講、教室変更などのユーザに必要な情報がホーム画面に表示。
- 教務課・学生課 Web ページ情報の配信
ホーム画面のお知らせからをタップで画面移動ができ、成績などの項目ごとに情報が配信。

NaNApp がリリースされたことで、受講している講義や講義情報をサービスにログインすることなく時間割の確認が取れるようになった。しかし問題点として、授業を各自で登録することが学生の負担になってしまう点、お知らせが配信がされなかった点、不具合が多発したことで運用に負担がかかり学生が引き継ぐことができなかった点が挙げられる。

3 マイクロサービスアーキテクチャを用いた学生情報システムの再設計

3.1 マイクロサービスアーキテクチャについて

マイクロサービスアーキテクチャ [3] とは、単一なアプリケーション小さいサービス群として組み合わせて構築されている。1つのアプリケーションに全ての機能が詰め込まれているモノリシックな構成のプログラムではなく、複数のサービスや機能を疎結合させ構築してきたものである。

マイクロサービスアーキテクチャには二つの特徴がある。一つ目は自律性 [4] という各コンポーネントサービスを、他のサービスの機能に影響を与えずに開発、デプロイ、実行、スケールすることができる特徴がある。次に特殊性 [4] は各サービスは一連の機能の為に設計され、特定の問題を解決することにフォーカスしている。開発者が経時的にコードをサービスに追加し、サービスが複雑になる場合、サービスをより小さなサービスに分割できる特徴を持っている。

3.2 マイクロサービスアーキテクチャの利点

全ての機能が単一のシステムとして動くモノリシック構成では、アプリケーションの拡大に伴って扱いにくくなり技術選択が困難であったのに対して、マイクロサービスアーキテクチャでは以下の利点が挙げられる [5]。

- サービス毎の技術特異性
- 回復性（レジリエンス）
- 組織面の一致
- デプロイの容易性
- スケーリングの効率化
- 合成可能性
- 交換可能にするための最適化

本研究で特に重要になるのは回復性とデプロイの容易性、交換可能にするための最適化であると考えられる。

回復性とは、システムのあるサービスに障害が発生した際に、障害が連鎖せず問題を分離してシステムの残りの部分は機能し続けることができる性質がある。

デプロイの容易性について、マイクロサービスでは一つ一つのサービスが独立しているので、コードの変更や障害の特定などに迅速に対応することができる。

交換可能にするための最適化について、モノリシックのシステムではある機能を交換、削除しようと思うと大規模なコードの変更が必要になることがある。マイクロサービスではそれにかかる手間を大幅に削減することが可能なことから運用にかかるコストを最適化することができる意味である。

これらの利点を踏まえて、本研究では開発するアプリケーションにマイクロサービスアーキテクチャの技術を用いて先に述べた課題を解決する。

3.3 統一化アプリケーションの設計

我々は以下の4つの要件を定義しアプリケーションを開発する。

1つ目は異なる学内 Web サービスが混在していることで、利用目的が違うサービスを横断的に利用しているのでは各サービスの情報を1つのアプリケーションで見られるようにする。

2つ目は登録している講義からのお知らせや PORTA の学生向け最新情報をホーム画面に配置することで学生がアプリを起動するだけで必要な情報が入手できるようにする。

3つ目は南山大学の学生および大学院生計 34 人から「学生生活で困っていることもしくは困っていたこと」、「南山生向けのアプリケーションが開発される場合、あなたのほしい機能」をアンケート集計した。特にスマートフォン端末上で提供することで学生の利便性が向上されると思われる機能と要望から検討した。検討結果、本研究ではキャンパスマップサービスを導入する。

4つ目は問い合わせフォームを作成し、ユーザに問題や不満があった際に問い合わせをもらい、それらを集計した上で迅速に修正できる環境を整える。

3.4 サービスの分割

マイクロサービスアーキテクチャにおけるサービスの分割には疎結合と高凝集性を考慮しなければならない。2つの特徴を踏まえ、本研究におけるサービスの分割は次のように考えた。表 1、図 1 は、それぞれのサービスがデータ取得するデータベース先を示している。例えば、お知らせサービスは PORTA、WebClass からデータを取得している。また、API ゲートウェイを経由することで複数のサービスの API を一括に管理でき、複雑な API 処理をもまとめることができる。

前述した4つの考えと分割したサービスを基に考案した画面遷移図は、図 2 の通りである。この図では、中心にホーム画面を配置している。ホーム画面からは他の5つの画面へ移動を考えている。

表 1 サービスの分割

サービス名	データの取得先
お知らせ	PORTA, WebClass
時間割	PORTA, シラバスデータベース
マップ	南山大学ホームページ
スケジュール	WebClass, 南山大学ホームページ
課題提出	WebClass, 講義資料ダウンロードサーバ

3.5 作成したサービスについて

本研究では、キャンパスマップサービスとその他ユーザ画面のサービスを作成した。キャンパスマップは、学内の棟の枚数分マップ写真を用いて棟のアルファベットを選択

することで選択した棟の場所が赤丸で示される。

次にその他ユーザの画面はブックマークのようにサイト項目をタップすることで学内 Web サービスの各サイトへの移動できることを考えている。また問い合わせフォームの開設をすることで、ユーザの不満や新機能導入のために意見を集めることを考えている。

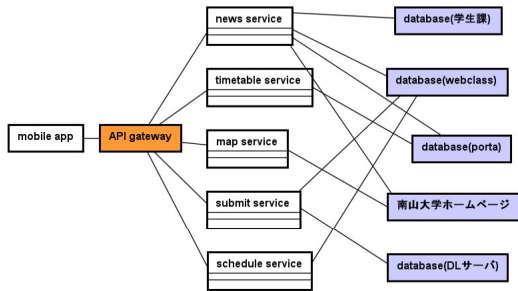


図1 サービス構成図

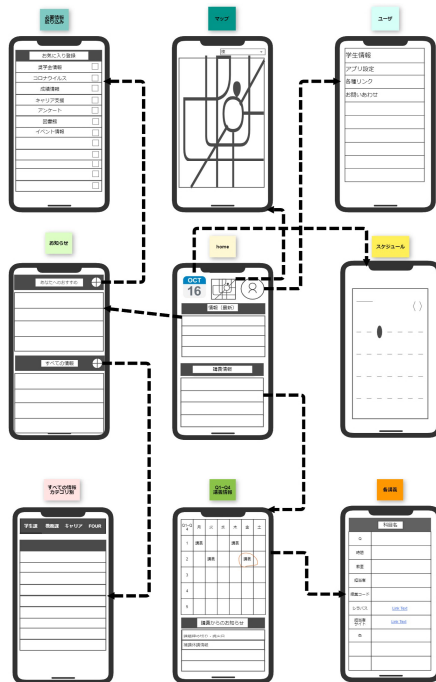


図2 アプリケーションのモデル画面遷移図

4 考察

4.1 既存アプリケーションとの比較

本節ではマイクロサービスアーキテクチャの技術を用いる統一化アプリケーションと NaNApp と比較することで提案手法の得失について考察する。

3章のマイクロサービスアーキテクチャの利点について

で述べた回復性、デプロイの容易性、交換可能にするための最適化について、NaNApp と本研究で設計した結果について比較をした。

4.1.1 回復性

NaNApp では一部の機能で障害が発生した際、原因の究明に時間を要することや、システム全体に影響をおよぼす可能性があるといった問題点がある。しかしマイクロサービスアーキテクチャを利用した設計を行うことで、NaNApp で頻発していたお知らせ機能の不具合が起きた際、お知らせ機能のみ使用できなくなりその他のサービスを引き続き提供することができる。

4.1.2 デプロイの容易性

NaNApp では機能の追加や変更を行う際に、一部のコード変更がシステム全体に影響を及ぼし動作しなくなる可能性がある。その結果、一部の変更に対して全体のデプロイが必要になる。これにより、新しい機能の実装や機能の変更には慎重な対応が必要になる。しかしマイクロサービスアーキテクチャを用いると、機能の追加や変更を行う際に、サービスごとが疎結合になっているので、他のサービスに影響を与えることなくデプロイが可能になる。これにより、新しい機能や機能の変更を気軽に行うことができる。その結果、アプリケーション内や SNS を利用して学生にレビューを求めることで学生の不満や負担になっていることを聞き出しアプリケーションに素早く反映することができる。

4.1.3 交換可能にするための最適化

NaNApp ではサービス規模の機能を変更、削除しようとした際に、コードが複雑に絡み合っていることで、コード変更にかかるコストが高くなる。しかし、マイクロサービスアーキテクチャを利用した設計だともともとサービスごとで分かれているので、あるサービスの変更に対して他のサービスに影響なく変更することができる。

4.2 提案手法の有効性

NaNApp は、モノリシックで構成されるアプリケーションであることから大規模なコードで構成されていた。全体を把握することは大規模なシステムに加え、他人が制作していることで複雑すぎて読み取ることが困難であった。そこで、時間割のサービスにだけ注目し NaNApp で利用されていたものを図3に示す。図3からわかるように NaNApp の時間割サービスは7つのファイルで成立していた。この時間割機能の7つのファイルはモノリシックである NaNApp の一部の機能であるので、いずれかのファイルコードを変更や追加した場合、全体に影響が出る可能性がある。次にマイクロサービスアーキテクチャの技術を用いた時間割サービスを図4に示す。この図では時間割のレイアウトとなるものを API ゲートウェイを経由してアプリケーションに呼び出している。またレイアウト内に表

示される時間割のデータを PORTA から取得することで NaNApp のように学生がアプリケーション内に時間割を再登録せずにできる。このことから本研究の設計を用いることで機能の構造を単純化することができる。

これらのことからマイクロサービスアーキテクチャを利用することで、機能の追加や変更に強いアプリケーションのソフトウェア基盤を構築することができる。その結果利用者からの要望を取り入れやすくなるので、学生の負担を軽減が可能になる。

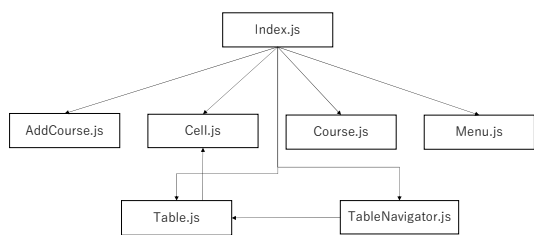


図 3 NaNApp における timetable

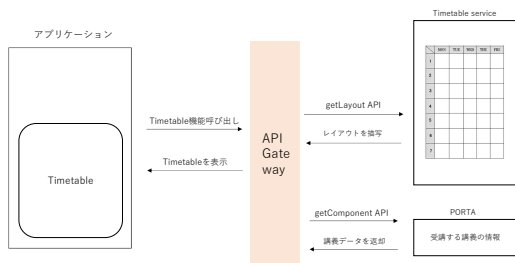


図 4 マイクロサービスアーキテクチャを用いた timetable

4.3 今後の課題

本研究では、学内 web サービスから API を利用してデータを取得することができなかったのでサービスの開発を進めることができなかった。今後は API の使用許可やデータの扱いについて考える必要がある。また、サービスを分割しすぎるとシステム全体が複雑化していき管理の対象が増加し運用管理がしづらくなる。今後もサービスの分割については継続してより適切なもの考える必要がある。

本研究で作成したキャンパスマップサービスについて、現在は変更があった際に手間のかかる設計になっているので、キャンパスマップのデータ取得や GPS 機能を利用してサービスの設計を見直す必要がある。

5 おわりに

情報システムが普及し大学では多岐にわたるサービスが Web を介して提供されている。南山大学では複数のサービスが存在することで学生に負担が生じているので、我々はマイクロサービスアーキテクチャの技術を用いて、各サービスを統一的に組み合わせたアプリケーションのソフトウェア基盤の構築を目指した。

提案手法の効果として、学内サービスを横断せずに利用できることで、毎回ログインする必要がなくなり、学生の必要な情報を手間なく見つけることで学生の負担が軽減できることを示した。学生の負担の軽減に加え、学内 web サービスの追加や変更があった際のアプリケーションの変更を容易にすることで、大規模なコード変更をすることなく必要最低限の修正が可能になったことが分かった。

NaNApp と提案するソフトウェア基盤に基づくアプリケーションを比較し、考察した。その結果、提案するソフトウェア基盤を利用することで、学生の負担を軽減するソフトウェア基盤を構築できることが示された。

今後の課題は、API 使用を前提に提案したが API の使用許可が下りなかったので API の使用許可、あるいは学生の個人データの扱いについて考える必要がある。また、本研究で作成したキャンパスマップサービスは、マップの情報などの変更があった際に手間のかかる設計になっているので、設計を見直す必要がある。マイクロサービスアーキテクチャの利点や NaNApp との比較から提案するアプリケーションをマイクロサービスアーキテクチャとして実際に開発していくことが重要と考えられる。

参考文献

- [1] 総務省, “第 2 部 基本データと政策動向”, <https://www.soumu.go.jp/johotsusintokei/whitepaper/ja/r02/html/nd252120.html>, 2020.
- [2] “南山大学在学生向けスマートフォンアプリの製作報告 (南山大学学生課)”, <https://office.nanzan-u.ac.jp/student-services/clubs/nanchare-hokoku/000182.html>, 2017
- [3] Sam Newman, 島田浩二 (訳), “モノリスからマイクロサービスへ”, 株式会社オライリー・ジャパン, 東京, pp. 4-9, 2020.
- [4] “マイクロサービスの概要 | AWS”, <https://aws.amazon.com/jp/microservices/>.
- [5] Sam Newman, 佐藤直生 (監訳), 木下哲也 (訳), “マイクロサービスアーキテクチャ”, 株式会社オライリー・ジャパン, 東京, pp. 5-10, 2016.