

看護学科におけるプログラミング教育を導入する試み - マタニティ支援アプリ開発を起点にした看護学生への情報教育

皆月 昭則[†] 西川 奏[‡]
釧路公立大学^{†‡}

1. はじめに

2020年に始まる小学校のプログラミング教育の話題に触れながら、非情報系の学生にプログラミングの学びの重要性を伝えているが、「なぜプログラミングを学ぶ必要があるのか」という質問が出される。質問の質を問うことはしないが、プログラミングに対しては、学ぶ「苦手」意識とは別の抵抗感がある。図1のように数学や英語は、苦手意識があっても、学ぶことへの抵抗感はないようであるが、プログラミングに対しては、特別な違和感があるようである。

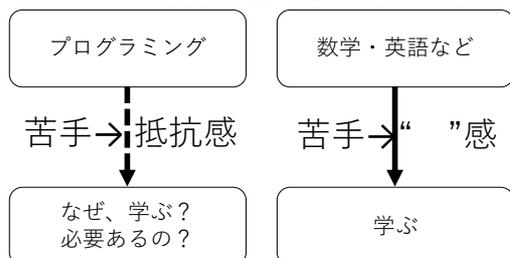


図1 プログラミングに対する苦手・抵抗感

看護学科の学生も同様で、プログラミングを学ぶ違和感は、初回の講義時から表出する。抵抗感なくプログラミングに興味関心を向けて、積極的に学ぶようにするのか、講義内容・展開と実習支援システム構築の長年の難題であった。試みでは、看護学科における情報科学という科目に、適切なプログラミング技術の教科書を選定して、Monaca という総合開発環境でプログラミングの実習を設定し、学生個別にアプリ製作が達成できる講義内容と実習の実践報告である。

2. マタニティ（妊婦）支援アプリのインパクト

講義の初回では、図2のマタニティ支援アプリを紹介した。非情報系の学生によるアプリ開発・改良による公開配布している動機感で、看護学生たちは驚いた。2017/2018年にわたり、アプリ開発に向き合い、妊婦の健康支援をしている活動のストーリーに、看護学生たちは傾聴して、プログラミング技術へ興味関心が高まった。

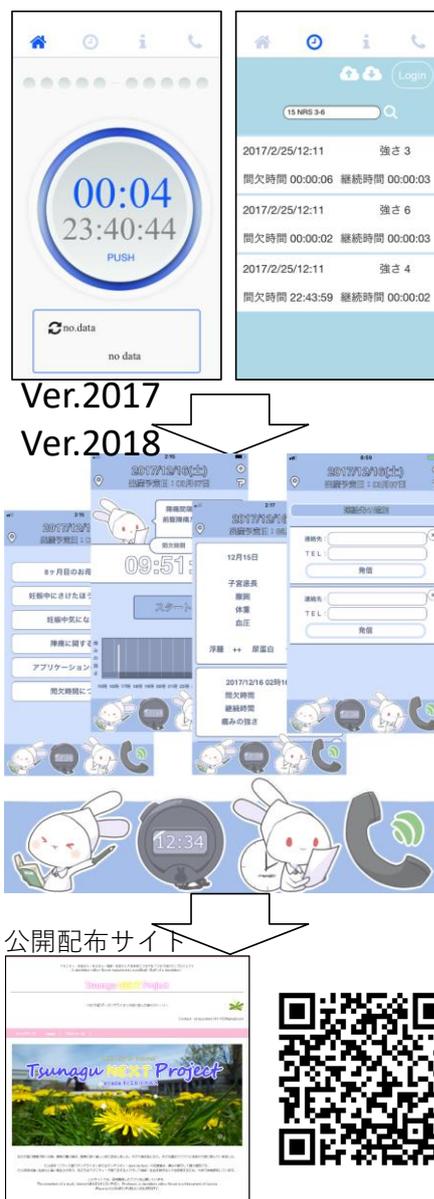


図2 動機付けに用いたマタニティ支援アプリ

3. Monaca を用いたプログラミング講義と学習

学生は Monaca 実習の開発支援環境（以下 MonacaIDE）を用いるプログラミング技術の教科書で、3種類の言語（HTML5, CSS3, JavaScript）を学び、実習用のコードを入力する12章構成と5つの付録である。90分の講義時間の配分は、60分が講義（教科書の技術解説は約20～30分、約30～40分を医療・看護情報の解説）で、残り

A Considering of Programming Education in Department of Nursing

[†]Minaduki Akinori · kushiro Public University

[‡]Nishikawa Kanade · kushiro Public University

の約 30 分を実習用 PC と学生各自のスマートフォンの使用を許可した実習である。実習の MonacaIDE の手順は図 3 に示す。1 講義の展開は 1 章の技術解説の後に実習のための URL を「インポート」してサンプル「コード」を入力。実行は PC の簡易的な「プレビュー」と学生各自のスマートフォンの接続で動作確認が可能である。



図 3 MonacaIDE による実習手順 (PC 画面)

3.1 学生各自のスマートフォンを実習で試す

看護情報学の参考書では「第 5 章：医療における情報システム」という章で、医療用画像管理システム (PACS) ・放射線部門業務システム (RIS) や病院情報システム (HIS) やクラウド型システムの解説を併用してプログラミング技術の講義展開した。Monaca の教科書の付録 3 には、カメラ機能のサンプルがインポート可能であり、通信に関する技術や写真撮影時の肖像権など、看護情報学の参考書「第 6 章・7 章・8 章：患者の権利と情報など」の解説を混合しながら講義展開が可能であった。



図 4 カメラ機能のプログラム実習の様子

3.2 サンプルアプリの関心から医療アプリ開発

3.1 節で述べたようなカメラ機能を応用した患者の褥瘡の状態記録の試み開発などプログラミングに積極的に取り組む学生や、Monaca 教科書の BMI アプリを参考に、看護学科で学んだ多くの知識・定式をタブレットに実装して、臨床実習で役立つアプリ開発に取り組む学生もいた。

4. 国家試験アプリの開発課題と作品発表

学生への評価 (100%) は、個別のアプリ作品を 20% の評価にして、80% を共通課題の作品の評価にした。共通課題は、看護師国家試験対策アプリを 2 種、MonacaIDE を提供するアシアル社からのインポートのサンプルコードを使用した。



図 5 共通課題に取り組む様子



図 6 制作した国家試験アプリと作品発表の様子

5. おわりに

講義や実習において、学生から「なぜプログラミングを学ぶ必要があるのか」という長年の問いに困惑している教育関係者も多い。プログラミング的論理思考や教養であるという強調でも、学生は、学ぶ必要性は感じてくれるかも知れないが、学びの積極性は得られない。プログラミングが小学校の必修化になることで、数学や英語科目のように「学ぶ」抵抗感は軽減されるかも知れないが、終わってみると微積は苦手、英会話は苦手という人々もいるような科目になってはいけない。看護学科での試みは、看護の専門性分野という「現地」の情報・知識を活かし、看護師の業務を支援する「現物」を作る (プロトタイプのアプリケーション設計) 積極性を期待し、臨床実習などで「現実」に使ってみて、問題抽出する仕掛けである。カイゼンを全面に進めるトヨタ方式をプログラミング教育に応用してみた。問題に気づき、モノづくりの設計目標・要件が明確になれば、実際の開発はシステムエンジニアやプロのプログラマーに任せるモノづくりの三現主義の共創志向性に依拠した方法論であり、今後、数年、この仕掛けを非情報系の学生に向けて試行する予定である。

参考文献

教養としてのプログラミング的思考 : 今こそ必要な「問題を論理的に解く」技術、草野俊彦、SBクリエイティブ、2018