

# Using a Microcomputer as a Standalone Specialist

Kim Yun\*

Department of Biomedical Engineering, University of Tokyo, Chome Hongo, Bunkyo City, Tokyo, Japan

## Introduction

The development of computer applications in the field of mental health has always placed more of a focus on meeting institutional information demands than on meeting the needs of individuals. The development of software to assist the lone practitioner has been fragmented and has lagged behind improvements in hardware cost performance ratios. Current microcomputer advancements have a foundation in the study conducted twenty years ago on the applicability of computer technology to institutional mental health information needs. Early computer programmes were designed to handle a wide range of administrative and clinical tasks for big mental health facilities, which produced a lot of record keeping and reporting demands. The latest generation of multi-user, 16- and 32-bit microcomputers are specifically engineered with more processing and storage capabilities for user flexibility and complexity. The mental health community has not yet benefited from these computer system capacity developments. Many, possibly most, minicomputer jobs as well as certain mainframe programmes can be replaced by the new machines. Microcomputers will be a professional's must-have equipment. The new processing power of microcomputers is based on a 16 and 32 bit architecture, allowing for quicker operation and higher numerical calculation precision. The new machines' expanded memory, which has up to 512 K internal storage and 10–20 megabytes of hard disc storage, is another benefit. The trend toward greater memory is still present [1,2].

## Description

A survey of Computers in Psychiatry/Psychology (CP/P) in the present shows that programmes used by mental health providers are neither standardised or integrated into effective software systems with regard to administrative and management tasks. The special administrative and management requirements of the mental health doctor or practitioner have not been taken into account when designing them. Instead, they are modifications of already-available software packages for word processing, spreadsheets, and database management. Office management apps have received negative reviews for their cost effectiveness in the few evaluations written by practitioners. These critiques, like many other evaluations, frequently require an update. A lone practitioner typically creates specialised mental health software programmes to match his unique practise demands. Analysis of numerous software directories and services shows that only a small number of these apps are provided commercially, and they are neither readily accessible nor well-publicized. The few programmes that do exist have typically not been portable across various operating systems or hardware setups. Typically, they have not undergone extensive testing before being put to use widely because doing so is expensive and necessitates significant R&D expenditure [3,4].

*\*Address for Correspondence:* Kim Yun, Department of Biomedical Engineering, University of Tokyo, Chome Hongo, Bunkyo City, Tokyo, Japan; E-mail: yun.kim@up.ac.za

**Copyright:** © 2022 Yun K. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Date of Submission:** 03 June, 2022, Manuscript No. GJTO-22-71127; **Editor Assigned:** 05 June, 2022, PreQC No. P-71127; **Reviewed:** 10 June, 2022, QC No. Q-71127; **Revised:** 15 June, 2022, Manuscript No. R-71127; **Published:** 20 June, 2022, DOI: 10.37421/2229-8711.2022.13.302

The development of clinical apps for the specific provider has moved slowly and haphazardly. The additional complexity of the requirements for clinical applications necessitates many person years of programming. According to their explanation, creating software for a niche market is "so time consuming, so expensive, that for programmers it is a doubtful endeavour." Only a small number of programmes have been created for microcomputer applications because of this. Programs for biofeedback, psychological testing, and patient assessment are a few examples. The creation of programmes with a single function and a lack of an integrated design and structure has been largely influenced by the difficulty and high expense of programming clinical applications. Furthermore, the design and functionality of these sophisticated applications are limited. There is now a tonne of software available or being developed for microcomputers. Applications from mini- and mainframe computers are being ported to microcomputers thanks to the rise in accessible memory for small computers. There are currently no reviews of new goods that manufacturers tout as "sophisticated, simple to use, and interactive," like the evaluation and case management software from Applied Innovations. However, these products suggest that such software for the mental health sector is on the way. In general, non-projective measurements are the main focus of computerised diagnostic applications. Projective exams and interviews should not be computerised for the following reasons: (1) relative difficulty in the administration of projective procedures; (2) ill-defined interpretive norms; and (3) overall clinician hostility to such computer-assisted approaches [5].

## Conclusion

In this thorough investigation of software networks built from Java source code, we investigate macroscopic network characteristics connected to the structural layout of the relevant software project. To identify the most important and susceptible software classes, we then do a microscopic node-level analysis of the networks. Finally, we examine mesoscopic network structural components and demonstrate how they might be used in project refactoring. We demonstrate, among other things, how difficult it is to regulate software systems despite the fact that they are very susceptible to processes like bug spread. Only 17 percent of the Java namespace, however, may be used to govern the Java language. Additionally, we include a number of network-based quality indicators that may be used to rate the design, reusability, robustness, controllability, and other aspects of software projects. The study reveals network analysis in this way.

## Acknowledgement

None.

## Conflict of Interest

The authors reported no potential conflict of interest.

## References

1. Levitan, Karen B., Elizabeth A. Willis and Jeffrey Vogelgesang. "Microcomputers and the individual practitioner: A review of the literature in psychology and psychiatry." *Comp Human Ser* 1 (1985): 65-84.
2. Sheingold, Karen, Janet Kane, and Mari Endrewit. "Microcomputer use in schools: Developing a research agenda." *Harvard Ed Rev* 53 (1983): 412-432.

3. Wilson, Ronald G and Duane L. Smith. "Microcomputer applications for primary health care in developing countries." *Infect Dis Clinics North Am* 5 (1991): 247-264.
4. Sieverts, Eric G and Marten Hofstede. "Software for information storage and retrieval tested, evaluated and compared Part 1-General introduction." *Elec Library* (1991).
5. Dologite, D.G. "Measuring microcomputer literacy." *J Ed Tech Sys* 16 (1987): 29-43.

**How to cite this article:** Yun, Kim. "Using a Microcomputer as a Standalone Specialist." *Glob J Tech Optim* 13 (2022): 302.