

Evidence-Based Practice for Designing Accessible Games: How Can Mainstream Games Accommodate for the Visually Impaired?

COMP160 - Software Engineering

1608351

March 21, 2017

Please include an abstract of at most 100 words (these do not count towards your word count).

1 Introduction

Developing games to meet the many varying cognitive, physical, and visual needs of the population is a complicated and often expensive task. Whilst other software, such as office and web applications, have been adapted successfully for the visually impaired, accessibility in the gaming industry is comparatively poor.

The reliance of mainstream games on impressive visuals makes adaptation for the visually impaired far more complex than other applications. However,

several creative technologies and strategies have been designed to improve games for the visually impaired, using music to signify transitions [?], echoes to describe 3D spaces [?], specially designed gaming devices [?] [?], and the conversion of images into sounds [?].

Within software engineering, developers often face tight deadlines which may prevent accessibility from being prioritized in the engineering of games [?] [?] . This paper aims to explore obstacles facing the development of accessible games and how mainstream games can learn from software designed specifically for the visually impaired.

2 Case Studies

AudioQuake:

Quake [?] is one of the first FPS mainstream games to be adapted to suit the visually impaired, in the form of AudioQuake, which uses an auditory interface [?]. The creators were keen to not only enable basic game play, but ensure blind users could participate online and in the building of their own level maps. Adapting Quake into an accessible game for the visually impaired involved several stages of development, beginning with the low-level requirements necessary to implement an auditory interface. Audio needed to be improved to create more realistic cues for the visually impaired, such as indicating room size using echoes. AudioQuake is part of the AGRIP project [?] which aim for games, to not only playable by the disabled, but that enables them to play evenly matched with or against able-bodied players, create their own map levels and adjust game settings. AudioQuake was built with LDL (Level Description Language), an XML programming language designed specifically to enable developers to describe 3D spaces and for visually impaired players to be able to create 3D levels. A vital part of the

AGRIP Project is community contribution, and particularly disabled-users feedback. During development, AudioQuake was regularly play-tested at conventions, such as Sight Village, and conversations between users on the mailing list were encouraged for greater insight into the game's effectiveness.

AccTrace:

Although considerable research and proposals for improving Software Engineering for accessibility exist, developers rarely understand how to code accessible systems. AccTrace [?] is a Case Tool designed to use an ontology, a method for showing a group of concepts and the relationships between them, to propose and define a technical procedure for implementing accessible gaming. AccTrace was also designed to enable traceability, allowing developers to ascertain the origin of certain elements within a systems design. [?]

A Walk in the Park:

3 Software Engineering for Accessibility in Mainstream Games

4 Obstacles to Accessible Software

1. Valuing of Best Practices in Software Engineering: in academia, time constraints/pressures, focus on product being delivered asap
2. Valuing of accessibility in software engineering, and specifically value of gaming for the visually impaired:
3. Financial and Time Constraints: 'the efforts of game accessibility must have a realistic financial grounding, otherwise they risk not become implemented in mainstream games.' 'to get mainstream games to be accessible to as many as possible we need first to resolve the financial issues, which

are related to the time and effort accessibility development takes, and the increased number of sales you get by doing it.” [?]

4. Evenly-matched games between able and non-able bodied players
5. Standardised Guidelines for Accessible Games:

5 Recommendations

1. Standardised Guidelines
2. User Testing/Feedback for target group
3. Multimodal User Interfaces: Not only to suit visually impaired but diverse range of needs
4. Allowing users to make modifications to game: Not just basic playable, enable evenly-matched play against able-bodied users and the ability to adjust map levels, game settings, personalized characters (through use of audio/voice rather than appearance)
5. Prioritization of Accessibility: Teaching of accessible game engineering best practices in higher-education, awareness of benefits of games for visually impaired.
6. Make use of and develop further available software engineering tools for accessibility: LDL, AccTrace
7. Games must still be games when adapted. Not just usable, but enjoyable

6 Conclusion

Application of accessibility during the development of games, and not only an afterthought

References

- [1] P. A. d. Oliveira, E. P. Lotto, A. G. D. Correa, L. G. G. Taboada, L. C. P. Costa, and R. D. Lopes, “Virtual stage: An immersive musical game for people with visual impairment,” in *2015 14th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, Nov 2015, pp. 135–141.
- [2] M. T. Atkinson and C. H. C. Machin, “Making the Mainstream Accessible: Redefining the Game,” *Sandbox Symposium*, 2006.
- [3] S. Ohtsuka, N. Sasaki, S. Hasegawa, and T. Harakawa, “A new game device using body-braille for visually impaired people,” in *2007 First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning (DIGITEL’07)*, March 2007, pp. 197–199.
- [4] S. Ohtsuka, N. Sasaki, and Y. Fukunaga, “A vibration watch using a mobile phone for visually impaired people,” in *The 1st IEEE Global Conference on Consumer Electronics 2012*, Oct 2012, pp. 316–317.
- [5] J. B. Marshall, G. Tyson, J. Llanos, R. M. Sanchez, and F. B. Marshall, “Serious 3d gaming research for the vision impaired,” in *2015 17th International Conference on E-health Networking, Application Services (HealthCom)*, Oct 2015, pp. 468–471.
- [6] B. Stroustrup, “What should we teach new software developers? Why?” *Communications of the ACM*, vol. 53, no. 1, p. 40, jan 2010. [Online]. Available: <http://portal.acm.org/citation.cfm?doid=1629175.1629192>
- [7] S. T. Frezza, “Issues in student valuing of software engineering best practices,” in *2016 IEEE Frontiers in Education Conference (FIE)*, Oct 2016, pp. 1–4.
- [8] id Software, “Quake,” [video game, Linux], 1996.

- [9] M. Atkinson and S. Gucukoglu, “Agrip project,” Online, 2003 - 2012.
[Online]. Available: <http://agrip.org.uk>
- [10] R. G. De Branco, M. I. Cagnin, and D. M. B. Paiva, “AccTrace: Accessibility in phases of requirements engineering, design, and coding software,” in *Proceedings - 14th International Conference on Computational Science and Its Applications, ICCSA 2014*, 2014, pp. 225–228.
- [11] K. Bierre, J. Chetwynd, B. Ellis, D. M. Hinn, S. Ludi, and T. Westin, “Game Not Over: Accessibility Issues in Video Games.”