10
References

42 Entertainment, 2007. the beast. Available at: http://www.42entertainment.com/beast.html [Accessed August 13, 2011].

Adams, E., 2010. *Fundamentals of game design*, New Riders.

Akenine-Möller, T. and Ström, J., 2008. Graphics Processing Units for Handhelds. *Proceedings of the IEEE*, 96(5), pp.779-789.

Ambler, S.W., 2005. *The Elements of UML™ 2.0 Style*, Cambridge University Press.

Anatel, 2011. *Relatório anual 2010*, Available at: http://www.anatel.gov.br/Portal/verificaDocumentos/documento.asp?numeroPublicacao=260639&assuntoPublicacao=Relat%F3rio%20Anual%202010&caminhoRel=Cidadao-Biblioteca-Acervo %20Documental&filtro=1&documentoPath=260639.pdf [Accessed August 16, 2011].

Ansca, 2011. Ansca Mobile's cross-platform mobile app development tool. Available at: http://www.anscamobile.com/corona/ [Accessed August 10, 2011].

Apple, 2011. Apple - iPhone 4 - Size, weight, battery life, and other specs. Available at: http://www.apple.com/iphone/specs.html [Accessed July 27, 2011].

Arango, G.F., 1988. *Domain engineering for software reuse*. PhD Thesis. University of California, Irvine.

Arduino Software, 2011. Arduino - HomePage. Available at: http://www.arduino.cc/ [Accessed August 6, 2011].

Ballagas, R. and Walz, S.P., 2007. REXplorer: Using Player-Centered Iterative Design Techniques for Pervasive Game Development. In C. Magerkurth & C. Roecker, eds. *Pervasive Gaming Applications – A Reader for Pervasive Gaming Research*. pp. 255-284.

Ballagas, R., Kratz, S.G., Borchers, J., Yu, E., Walz, S.P., Fuhr, C.O., Hovestadt, L. and Tann, M., 2007. REXplorer: a mobile, pervasive spell-casting game for tourists. *CHI '07 extended abstracts on Human factors in computing systems*, pp.1929–1934.

Ballagas, R., Kuntze, A. and Walz, S., 2008. Gaming Tourism: Lessons from Evaluating REXplorer, a Pervasive Game for Tourists. In J. Indulska, D. Patter-

son, T. Rodden, & M. Ott, eds. *Pervasive Computing*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 244-261. Available at: http://dx.doi.org/10.1007/978-3-540-79576-6_15.

Barkhuus, L., Chalmers, M., Tennent, P., Hall, M., Bell, M., Sherwood, S. and Brown, B., 2005. Picking Pockets on the Lawn: The Development of Tactics and Strategies in a Mobile Game. In M. Beigl, S. Intille, J. Rekimoto, & H. Tokuda, eds. *UbiComp 2005: Ubiquitous Computing*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 903-903. Available at: http://dx.doi.org/10.1007/11551201_21.

Bell, M. *et al.*, 2006. Interweaving mobile games with everyday life. *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pp.417–426.

Benford, S., Crabtree, A., Flintham, M., Drozd, A., Anastasi, R., Paxton, M., Tandavanitj, N., Adams, M. and Row-Farr, J., 2006a. Can you see me now? *ACM Transactions on Computer-Human Interaction (TOCHI)*, 13, pp.100–133.

Benford, S., Crabtree, A., Reeves, S., Sheridan, J., Dix, A., Flintham, M. and Drozd, A., 2006b. The Frame of the Game: Blurring the Boundary between Fiction and Reality in Mobile Experiences. *Proceedings of the SIGCHI conference on Human Factors in computing systems*, pp.427–436.

Benford, S., Flintham, M., Drozd, A., Anastasi, R., Adams, M., Row-Farr, J., Oldroyd, A., Sutton, J. and Park, A., 2004a. Uncle Roy All Around You: Implicating the city in a location-based performance. In *ACE04*. ACM Press. Available at: http://www.performancestudies.pl/dydaktyka/files/ace2004.pdf.

Benford, S., Magerkurth, C. and Ljungstrand, P., 2005. Bridging the physical and digital in pervasive gaming. *Communications of the ACM*, 48, pp.54–57.

Benford, S. *et al.*, 2004b. The Error of Our Ways: The Experience of Self-Reported Position in a Location-Based Game. In N. Davies, E. Mynatt, & I. Siio, eds. *UbiComp 2004: Ubiquitous Computing*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 70-87. Available at: http://dx.doi.org/10.1007/978-3-540-30119-6_5.

Bishop, R.H., 2007. *Mechatronic Systems, Sensors, and Actuators: Fundamentals and Modeling*, CRC Press.

Björk, S., Falk, J., Hansson, R. and Ljungstrand, P., 2001. Pirates! Using the physical world as a game board. In *Proceedings of Interact 2001*. IOS Press, pp. 9–13.

Björk, S., Holopainen, J., Ljungstrand, P. and Mandryk, R., 2002. Special Issue on Ubiquitous Games. *Personal and Ubiquitous Computing*, 6, pp.358–361.

Blanchette, J. and Summerfield, M., 2008. *C++ GUI Programming with Qt 4 (2nd Edition)* 2nd ed., Prentice Hall.

Broll, G. and Benford, S., 2005. Seamful Design for Location-Based Mobile Games. In F. Kishino, Y. Kitamura, H. Kato, & N. Nagata, eds. *Entertainment Computing - ICEC 2005*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 155-166. Available at: http://dx.doi.org/10.1007/11558651_16.

Bystrom, K.-E., Barfield, W. and Hendrix, C., 1999. A Conceptual Model of the Sense of Presence in Virtual Environments. *Presence: Teleoperators and Virtual Environments*, 8(2), pp.241-244.

Caillois, R., 2001. *Man, Play and Games*, University of Illinois Press.

Calvillo-Gámez, E.H., Cairns, P. and Cox, A.L., 2010. Assessing the Core Elements of the Gaming Experience. In R. Bernhaupt, ed. *Evaluating User Experience in Games*. London: Springer London, pp. 47-71. Available at: http://www.springerlink.com/content/vn18m5p67t216354/ [Accessed September 29, 2011].

Capin, T., Pulli, K. and Akenine-Moller, T., 2008. The State of the Art in Mobile Graphics Research. *IEEE Computer Graphics and Applications*, 28(4), pp.74-84.

Capra, M., Radenkovic, M., Benford, S., Oppermann, L., Drozd, A. and Flintham, M., 2005. The multimedia challenges raised by pervasive games. *Proceedings of the 13th annual ACM international conference on Multimedia*, pp.89–95.

Chalmers, M. and Galani, A., 2004. Seamful interweaving: heterogeneity in the theory and design of interactive systems. *Proceedings of the 5th conference on Designing interactive systems: processes, practices, methods, and techniques*, pp.243–252.

Chehimi, F., Coulton, P. and Edwards, R., 2008. Using a camera phone as a mixed-reality laser cannon. *International Journal of Computer Games Technology*, 2008, pp.2:1–2:6.

Chung, L. and Leite, J.C.S. do P., 2009. On Non-Functional Requirements in Software Engineering. In A. T. Borgida, V. K. Chaudhri, P. Giorgini, & E. S. Yu, eds. *Conceptual Modeling: Foundations and Applications*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 363-379. Available at: http://www.springerlink.com/content/l8p285p273461j60/ [Accessed June 25, 2011].

Chung, L., Nixon, B.A., Yu, E. and Mylopoulos, J., 1999. *Non-Functional Requirements in Software Engineering* 1st ed., Springer.

Cockburn, A., 2000. *Writing Effective Use Cases* 1st ed., Addison-Wesley Professional.

Cowley, B., Charles, D., Black, M. and Hickey, R., 2008. Toward an understanding of flow in video games. *Computers in Entertainment*, 6, p.1.

Crawford, C., 2003. *Chris Crawford on Game Design* 1st ed., New Riders Games.

Crawford, C., 1984. *The Art Of Computer Game Design: Reflections Of A Master Game Designer*, Osborne/McGraw-Hill,U.S.

Csikszentmihalyi, M., 1990. *Flow (The Psychology of Optimal Experience)* Book Club Edition (BCE)., Harper & Row.

Czarnecki, K., 1999. *Generative programming - principles and techniques of software engineering based on automated configuration and fragment-based component models*. PhD Thesis. Technical University of Ilmenau.

Davidsson, O., Peitz, J. and Björk, S., 2004. *Game design patterns for mobile games*, Nokia Research Center.

Davies, H., 2007. Place as media in pervasive games. *Proceedings of the 4th Australasian conference on Interactive entertainment*, pp.7:1–7:4.

Davis, S.B., Moar, M., Cox, J., Riddoch, C., Cooke, K., Jacobs, R., Watkins, M., Hull, R. and Melamed, T., 2005. 'Ere be dragons: an interactive artwork. *Proceedings of the 13th annual ACM international conference on Multimedia*, pp.1059–1060.

Dey, A.K., 2001. Understanding and Using Context. *Personal and Ubiquitous Computing.*, 5(1), pp.4–7.

Distimo, 2011. Distimo. Available at: http://www.distimo.com/ [Accessed October 8, 2011].

Distimo, 2010. *Full year 2010*, Available at: http://report.distimo.com/.

Drozd, A., Benford, S., Tandavanitj, N., Wright, M. and Chamberlain, A., 2006. Hitchers: Designing for Cellular Positioning. In P. Dourish & A. Friday, eds. *UbiComp 2006: Ubiquitous Computing*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 279-296. Available at: http://dx.doi.org/10.1007/11853565_17.

Egenfeldt-Nielsen, S., 2007. Third Generation Educational Use of Computer Games. *Journal of Educational Multimedia and Hypermedia*, 16(3), pp.263-281.

Ekman, I., Ermi, L., Lahti, J., Nummela, J., Lankoski, P. and Mäyrä, F., 2005. Designing sound for a pervasive mobile game. *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pp.110–116.

ESA, 2011. *Essential Facts about the computer and video game industry - sales, demographic and usage data*, Available at: http://www.theesa.com/facts/pdfs/ESA_EF_2011.pdf.

Flintham, M., Giannachi, G., Benford, S. and Adams, M., 2007a. Day of the figurines: supporting episodic storytelling on mobile phone. *Proceedings of the 4th in-*

ternational conference on Virtual storytelling: using virtual reality technologies for storytelling, pp.167–175.

Flintham, M. *et al.*, 2007b. Day of the figurines: A slow narrative-driven game for mobile phones using text messaging. In *PerGames 2007, 4th International Symposium on Pervasive Gaming Applications*. International Symposium on Pervasive Gaming Applications (PerGames) 2007. pp. 43-52.

Gamma, E., Helm, R., Johnson, R. and Vlissides, J., 1995. *Design patterns: elements of reusable object-oriented software*, Addison-Wesley Longman Publishing Co., Inc.

Gbanga, 2010. Gbanga Famiglia. *Gbanga, the game that is virtually everywhere.* Available at: http://gbanga.com/ [Accessed April 6, 2011].

Gigantic Mechanic, 2010. Gigaputt:: Make the world your golf course! Available at: http://giganticmechanic.com/games_gigaputt.html [Accessed April 12, 2011].

Goel, V. and Pirolli, P., 1989. Motivating the Notion of generic design within information-processing theory: the design problem space. *AI Magazine*, 10, pp.18–35.

Graessle, P., Baumann, H. and Baumann, P., 2005. *UML 2.0 in Action: A project-based tutorial: A detailed and practical walk-through showing how to apply UML to real world development projects*, Packt Publishing.

Graft, K., 2010. EA's Gibeau: Over-Annualization, Lack Of Innovation Led To Medal Of Honor's Downfall. *Gamasutra*. Available at: http://www.gamasutra.com/view/news/28367/EAs_Gibeau_OverAnnualization_Lack_Of_Innovation_Led_To_Medal_Of_Honors_Downfall.php [Accessed July 30, 2011].

Guo, H., Trætteberg, H., Wang, A.I. and Zhu, M., 2010. TeMPS: A Conceptual Framework for Pervasive and Social Games. In *Digital Game and Intelligent Toy Enhanced Learning, IEEE International Workshop on*. Los Alamitos, CA, USA: IEEE Computer Society, pp. 31-37.

Heitmeyer, C., Kirby, J., Labaw, B. and Bharadwaj, R., 1998. SCR: A toolset for specifying and analyzing software requirements. In A. J. Hu & M. Y. Vardi, eds. *Computer Aided Verification*. Berlin/Heidelberg: Springer-Verlag, pp. 526-531. Available at: http://www.springerlink.com/content/52u167356468q252/ [Accessed August 6, 2011].

Hinske, S., Lampe, M., Magerkurth, C. and Röcker, C., 2007. Classifying Pervasive Games: On Pervasive Computing and Mixed Reality C. Magerkurth & Ce. Röcker, eds. *Concepts and technologies for Pervasive Games A Reader for Pervasive Gaming Research*, 1, pp.11–38.

Hudson, A., 2011. Have computer games become boring? *BBC*. Available at: http://news.bbc.co.uk/2/hi/programmes/click_online/9538690.stm [Accessed July 30, 2011].

Huizinga, J., 1949. *Homo ludens: a study of the play-element in culture*, Routledge & K. Paul.

Huizinga, J., 1971. *Homo Ludens: A Study of the Play-Element in Culture*, Beacon Press.

IBGE, 2011. IBGE Censo 2010. Available at: http://www.censo2010.ibge.gov.br/primeiros_dados_divulgados/index.php [Accessed April 10, 2011].

IBM Research, 1998. The Convenience of Small Devices: How Pervasive Computing Will Personalize E-Business. *IBM Think Research*. Available at: http://domino.watson.ibm.com/comm/wwwr_thinkresearch.nsf/pages/bergman398 .html [Accessed May 25, 2011].

IPerG, 2008. Integrated Project of Pervasive Games. Available at: http://www.pervasive-gaming.org/index.php [Accessed July 26, 2011].

Ishii, H. and Ullmer, B., 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp.234–241.

Jakl, A., 2004. The Journey - A mobile location based adventure. Available at: http://journey.mopius.com/ [Accessed April 6, 2011].

Joffe, B., 2007. Mogi. In F. Borries, S. P. Walz, & M. Böttger, eds. *Space Time Play*. Birkhäuser Basel, pp. 224-225. Available at: http://dx.doi.org/10.1007/978-3-7643-8415-9_80.

Joffe, B., 2005. Mogi: Location and presence in a pervasive community game.

Jones, E., 2010. The Effect of Sequelitis on Interactive Narratives. Available at: http://www.gamasutra.com/blogs/EvanJones/20101128/6522/The_Effect_of_Sequelitis_on_Interactive_Narratives.php [Accessed August 10, 2011].

Juul, J., 2003. The game, the player, the world: looking for a heart of gameness. In C. Marinka & R. Joost, eds. *Level Up Conference Proceedings: Proceedings of the 2003 Digital Games Research Association Conference*. Utrecht: University of Utrecht, pp. 30-45. Available at: http://www.digra.org:8080/Plone/dl/display_html?chid=05163.50560.pdf.

Korhonen, H. and Koivisto, E.M.I., 2007. Playability heuristics for mobile multi-player games. *Proceedings of the 2nd international conference on Digital interactive media in entertainment and arts*, pp.28–35.

Korhonen, H., Saarenpää, H. and Paavilainen, J., 2008. Pervasive Mobile Games — A New Mindset for Players and Developers. In *Proceedings of the 2nd International Conference on Fun and Games*. Berlin, Heidelberg: Springer-Verlag, pp. 21–32. Available at: http://dx.doi.org/10.1007/978-3-540-88322-7_3.

LAC, 2011. MoCA Monitors. Available at: http://www.lac.inf.puc-rio.br/moca/monitor.html [Accessed July 21, 2011].

Lakoff, G., 1987. *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind* 1st ed., University of Chicago Press.

Lankoski, P., Heliö, S., Nummela, J., Lahti, J., Mäyrä, F. and Ermi, L., 2004. A case study in pervasive game design: the songs of north. *Proceedings of the third Nordic conference on Human-computer interaction*, pp.413–416.

Larman, C., 2004. *Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development* 3rd ed., Prentice Hall.

Leite, J.C.S. do P., Hadad, G.D.S., Doorn, J.H. and Kaplan, G.N., 2000. A Scenario Construction Process. *Requirements Engineering*, 5(1), pp.38-61.

Lindt, I., Ohlenburg, J., Pankoke-Babatz, U. and Ghellal, S., 2007. A report on the crossmedia game epidemic menace. *Computers in Entertainment (CIE)*, 5.

Lindt, I., Ohlenburg, J., Pankoke-Babatz, U., Ghellal, S., Oppermann, L. and Adams, M., 2005. Designing Cross Media Games. In *Proceedings of 2nd international workshop on pervasive gaming applications*. PerGames 2005. Munich, pp. 8-13.

Lindt, I., Ohlenburg, J., Pankoke-Babatz, U., Prinz, W. and Ghellal, S., 2006. Combining multiple gaming interfaces in epidemic menace. *CHI '06 extended abstracts on Human factors in computing systems*, pp.213–218.

Linner, D., Kirsch, F., Radusch, I. and Steglich, S., 2005. Context-aware Multimedia Provisioning for Pervasive Games. In *Multimedia, International Symposium on*. Los Alamitos, CA, USA: IEEE Computer Society, pp. 60-68.

Lombard, M. and Ditton, T., 1997. At the Heart of It All: The Concept of Presence. *Journal of Computer-Mediated Communication*, 3(2), pp.0-0.

Lyytinen, K. and Yoo, Y., 2002. Issues and Challenges in Ubiquitous Computing. *Communications of the ACM*, 45, pp.62–65.

MacLean, K.E., Snibbe, S.S. and Levin, G., 2000. Tagged handles: merging discrete and continuous manual control. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. CHI '00. The Hague, The Netherlands: ACM, pp. 225–232.

Magerkurth, C., Cheok, A.D., Mandryk, R.L. and Nilsen, T., 2005. Pervasive games: bringing computer entertainment back to the real world. *Computers in Entertainment (CIE)*, 3, pp.4–4.

Mann, S., 2001. Wearable computing: toward humanistic intelligence. *IEEE Intelligent Systems*, 16(3), pp.10- 15.

McGonigal, J., 2003. A Real Little Game: The Pinocchio Effect in Pervasive Play. In C. Marinka & R. Joost, eds. *Level Up Conference Proceedings: Proceedings of the 2003 Digital Games Research Association Conference*. Utrecht: University of Utrecht, p. CD ROM. Available at: http://www.digra.org/dl/display_html?chid=05097.11067.pdf.

McGonigal, J., 2006. *This might be a game: ubiquitous play and performance at the turn of the twenty-first century*. Berkeley, CA, USA: University of California at Berkeley.

Mernik, M., Heering, J. and Sloane, A.M., 2005. When and how to develop domain-specific languages. *ACM Computing Surveys*, 37(4), pp.316-344.

Montola, M., Stenros, J. and Wærn, A., 2009. *Pervasive Games: Theory and Design*, Morgan Kaufmann.

Montola, M., Waern, A. and Nieuwdorp, E., 2006. *Domain of pervasive gaming*, IPerG. Available at: http://www.pervasive-gaming.org/Deliverables/D5.3b-Domain-of-Pervasive-Gaming.pdf.

Motorola, 2011. Motorola ATRIX™ ME860, MB861 specification. Available at: http://developer.motorola.com/products/atrix-me860/ [Accessed July 27, 2011].

Murdley, C., 2006. Video Games are Dead: A Chat with Storytronics Guru Chris Crawford. *Gamasutra*. Available at: http://www.gamasutra.com/view/feature/2722/video_games_are_dead_a_chat_with_.php [Accessed July 29, 2011].

Nieuwdorp, E., 2007. The pervasive discourse: an analysis. *Computers in Entertainment (CIE)*, 5.

Nokia, 2011. Qt — Qt - A cross-platform application and UI framework. Available at: http://qt.nokia.com/products [Accessed July 21, 2011].

Nokia Developer, 2011a. Device Details -- Nokia N8-00. Available at: https://www.developer.nokia.com/Devices/Device_specifications/N8-00/ [Accessed July 27, 2011].

Nokia Developer, 2011b. Mobile game design. Available at: http://www.developer.nokia.com/Resources/Library/Design_and_UX/designing-different-applications/game-design/mobile-game-design.html [Accessed August 9, 2011].

Nokia Developer, 2010. Nokia Developer - Ovi Store statistics. Available at: http://www.developer.nokia.com/Distribute/Ovi_Store_statistics.xhtml [Accessed May 22, 2011].

OMG, 2011. *UML Superstructure Specification, v2.4*, Available at: http://www.omg.org/spec/UML/2.4/Superstructure/Beta2/PDF.

Orbster, 2008. GPS Mission - Mobile GPS Game. Available at: http://gpsmission.com/ [Accessed April 6, 2011].

Oxford English Dictionary, 2011. pervasive, adj. *OED Online*. Available at: http://www.oed.com/view/Entry/141666?redirectedFrom=pervasive#eid [Accessed May 23, 2011].

Paavilainen, J., Korhonen, H., Saarenpää, H. and Holopainen, J., 2009. Player Perception of Context Information Utilization in Pervasive Mobile Games. In A. Barry, K. Helen, & K. Tanya, eds. *Breaking New Ground: Innovation in Games, Play, Practice and Theory: Proceedings of the 2009 Digital Games Research Association Conference*. London: Brunel University. Available at: http://www.digra.org/dl/display_html?chid=09287.55066.pdf.

Parnas, D.L. and Madey, J., 1995. Functional documents for computer systems. *Science of Computer Programming*, 25(1), pp.41-61.

Peitz, J., Saarenpää, H. and Björk, S., 2007. Insectopia: exploring pervasive games through technology already pervasively available. In *Proceedings of the international conference on Advances in computer entertainment technology*. ACE '07. New York, NY, USA: ACM, pp. 107–114.

Peterson, J.L., 1981. *Petri Net Theory and the Modeling of Systems*, Prentice Hall PTR.

Pilone, D. and Pitman, N., 2005. *UML 2.0 in a Nutshell (In a Nutshell)* 2nd ed., O'Reilly Media.

Qt, 2011. Qt 4.7: Signals & Slots. Available at: http://doc.trolltech.com/4.7/signalsandslots.html [Accessed July 27, 2011].

Qt Mobility, 2011. Qt Mobility 1.2: Sensors. Available at: http://doc.qt.nokia.com/qtmobility-1.2/sensors-api.html [Accessed July 27, 2011].

Rashid, O., Bamford, W., Coulton, P., Edwards, R. and Scheible, J., 2006. PAC-LAN: mixed-reality gaming with RFID-enabled mobile phones. *Computers in Entertainment (CIE)*, 4.

Ravaja, N., Saari, T., Turpeinen, M., Laarni, J., Salminen, M. and Kivikangas, M., 2006. Spatial Presence and Emotions during Video Game Playing: Does It Matter with Whom You Play? *Presence: Teleoperators and Virtual Environments*, 15(4), pp.381-392.

Retaux, X., 2003. Presence in the environment: theories, methodologies and applications to video games. *PsychNology Journal*.

Retronyms, 2009. Seek 'n Spell: GPS Word Game for iPhone. Available at: http://www.seeknspell.com/ [Accessed April 6, 2011].

Rogers, Y. and Muller, H., 2006. A framework for designing sensor-based interactions to promote exploration and reflection in play. *International Journal of Human-Computer Studies*, 64(1), pp.1-14.

Rollings, A. and Adams, E., 2003. *Andrew Rollings and Ernest Adams on Game Design* 1st ed., New Riders Games.

Saarenpää, H., Korhonen, H. and Paavilainen, J., 2009. Asynchronous gameplay in pervasive multiplayer mobile games. *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pp.4213–4218.

Salen, K. and Zimmerman, E., 2004. *Rules of play: game design fundamentals*, MIT Press.

Sas, C., 2005. Sense of presence. *Encyclopedia of Human Computer Interaction*, pp.511-517.

Scheible, J., Tuulos, V.H. and Ojala, T., 2007. Story Mashup: design and evaluation of novel interactive storytelling game for mobile and web users. *Proceedings of the 6th international conference on Mobile and ubiquitous multimedia*, pp.139–148.

Schmidt, A., 2000. Implicit human computer interaction through context. *Personal Technologies*, 4(2-3), pp.191-199.

Schmidt, A., Aidoo, K.A., Takaluoma, A., Tuomela, U., Laerhoven, K. and Velde, W., 1999. Advanced Interaction in Context. In H.-W. Gellersen, ed. *Handheld and Ubiquitous Computing*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 89-101. Available at: http://www.springerlink.com/content/f0d3d5a3ec9enb1r/ [Accessed July 15, 2011].

Schneider, J. and Kortuem, G., 2001. How to Host a Pervasive Game - Supporting Face-to-Face Interactions in Live-Action Roleplaying. In *Interactions in Live-Action Roleplaying. UbiComp workshop on Designing Ubiquitous Computing Games*.

Slater, M., 1999. Measuring Presence: A Response to the Witmer and Singer Presence Questionnaire. *Presence: Teleoperators and Virtual Environments*, 8(5), pp.560-565.

Slater, M. and Wilbur, S., 1997. A framework for immersive virtual environments (FIVE) - Speculations on the role of presence in virtual environments. *Presence Teleoperators Virtual Environments*, 6(6), pp.603-616.

Sotamaa, O., 2002. All The World's A Botfighter Stage: Notes on Location-based Multi-User Gaming. In M. Frans, ed. *Computer Games and Digital Cultures Conference Proceedings*. Tampere: Tampere University Press, p. 10. Available at: http://www.digra.org/dl/display_html?chid=05164.14477.pdf.

de Souza, C.S., 2005. *The semiotic engineering of human-computer interaction*, MIT Press.

Spielberg, S., 2001. *A.I. Artificial Intelligence*, Warner Bros.

Struppek, M. and Willis, K.S., 2007. Botfighters. In F. Borries, S. P. Walz, & M. Böttger, eds. *Space Time Play*. Birkhäuser Basel, pp. 226-227. Available at: http://dx.doi.org/10.1007/978-3-7643-8415-9_81.

Szulborski, D., 2007. The Beast. In F. Borries, S. P. Walz, & M. Böttger, eds. *Space Time Play*. Birkhäuser Basel, pp. 228-229. Available at: http://dx.doi.org/10.1007/978-3-7643-8415-9_82.

Tuulos, V., Scheible, J. and Nyholm, H., 2007. Combining Web, Mobile Phones and Public Displays in Large-Scale: Manhattan Story Mashup. In A. LaMarca, M. Langheinrich, & K. Truong, eds. *Pervasive Computing*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 37-54. Available at: http://dx.doi.org/10.1007/978-3-540-72037-9_3.

Unity Technologies, 2011. UNITY: Game Development Tool. Available at: http://unity3d.com/ [Accessed August 10, 2011].

Valente, L., 2007. *Accelerinvaders v0.4 (with auto-fire and new moves)*, Available at: http://www.youtube.com/watch?v=uzSzh1t9_CA [Accessed July 30, 2011].

Valente, L. and Feijó, B., 2011. Cooperative non-visual games for mobile phones. *Computers in Entertainment Interactive Website*, (to appear), p.15.

Valente, L. and Ftylitakis, N., 2011. QBluetooth library project. Available at: https://projects.developer.nokia.com/qbluetooth [Accessed July 21, 2011].

Valente, L., de Souza, C.S. and Feijó, B., 2008. An exploratory study on non-visual mobile phone interfaces for games. In *Proceedings of the VIII Brazilian Symposium on Human Factors in Computing Systems*. IHC '08. Porto Alegre, Brazil, Brazil: Sociedade Brasileira de Computação, pp. 31–39. Available at: http://portal.acm.org/citation.cfm?id=1497470.1497475 [Accessed March 22, 2011].

Valente, L., Souza, C.S. de and Feijó, B., 2009. Turn off the graphics: designing non-visual interfaces for mobile phone games. *Journal of the Brazilian Computer*

Society, 15(1). Available at: http://www.scielo.br/scielo.php?pid=S0104-65002009000100005&script=sci_arttext&tlng=en [Accessed January 29, 2011].

Viterbo, J., Sacramento, V., Rocha, R., Baptista, G., Malcher, M. and Endler, M., 2008. A Middleware Architecture for Context-Aware and Location-Based Mobile Applications. *Proceedings of the 2008 32nd Annual IEEE Software Engineering Workshop*, pp.52–61.

Volkswagen, 2009. The World's Deepest Bin | The Fun Theory. Available at: http://www.thefuntheory.com/worlds-deepest-bin [Accessed May 29, 2011].

Waern, A., Montola, M. and Stenros, J., 2009. The three-sixty illusion: designing for immersion in pervasive games. *Proceedings of the 27th international conference on Human factors in computing systems*, pp.1549–1558.

Walther, B., 2005. Notes on the Methodology of Pervasive Gaming. In F. Kishino, Y. Kitamura, H. Kato, & N. Nagata, eds. *Entertainment Computing - ICEC 2005*. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 488-495. Available at: http://dx.doi.org/10.1007/11558651_47.

Weiser, M., 1991. The computer for the 21st century. *Scientific American*, 265(3), pp.66–75.

Wikipedia, 2011a. Capture the flag. Available at: http://en.wikipedia.org/wiki/Capture_the_flag [Accessed September 25, 2011].

Wikipedia, 2011b. Majestic (video game) - Wikipedia, the free encyclopedia. Available at: http://en.wikipedia.org/wiki/Majestic_(video_game) [Accessed May 26, 2011].

Wikipedia, 2011c. Pac-man. *Wikipedia*. Available at: http://en.wikipedia.org/wiki/Pac-man [Accessed May 24, 2011].

Wikipedia, 2011d. Tag (game). Available at: http://en.wikipedia.org/wiki/Tag_games [Accessed September 28, 2011].

Witmer, B.G. and Singer, M.J., 1998. Measuring Presence in Virtual Environments: A Presence Questionnaire. *Presence: Teleoperators and Virtual Environments*, 7(3), pp.225-240.

YD Online, 2011. GEO Hunters for iPhone, iPod touch, and iPad on the iTunes App Store. Available at: http://itunes.apple.com/us/app/geo-hunters/id417926906?mt=8&ign-mpt=uo%3D6 [Accessed May 28, 2011].

Appendix A
Pervasive games descriptions

This appendix describes briefly all games that have been analyzed as part of this research, to avoid cluttering the other chapters with the game descriptions. The games are sorted by year.

As this appendix works as a general reference, some game descriptions have been extracted from the original papers that describe the games. For all games, some properties have been highlighted: which sensors the game use, which network technology the game uses (if any), if the game requires a support staff while the game is happening, and other miscellaneous remarks that we found important to include.

A support staff is a team of people that monitor the game in the background while it is happening, in order to foresee and correct issues that might interrupt the game. This is typical of event games, and is referred in the literature of pervasive games as "orchestration" For examples on orchestration, please see (Benford *et al.* 2006a).

Some games use the mobile operator network (GPRS, EDGE, 3G) for accessing remote data. In these cases, the "network" property has been indicated as "operator".

A.1
Pervasive Clue (2001)

Sensors	:	proximity (RF sensors and beacons)
Network	:	-
Support staff	:	-
Others	:	-

The first game labeled as "pervasive", is a live-action role-playing game inspired on the board game *Clue* by Hasbro. In this game, players act as detectives

trying to solve mysterious murders. They have to find several clues to be able to solve the mystery.

The players use a PDA equipped with RF readers, and a magnifying glass. The magnifying class has cosmetic nature and the game uses this to transform the PDA into the "cluefinder", a way to make the PDA look more like a "game object" . The game uses physical objects to represent clues (knives, books, etc.). The objects are equipped with RF beacons, so a player is able to capture a clue by getting closer than 1 foot from the object.

A.2
Pirates! (2001)

Sensors	:	proximity sensors (short-range RF sensors)
Network	:	WiFi
Support staff	:	-
Other	:	requires place configuration

This is a game where players are captains of their ships, and have to sail around visiting islands, collecting resources, and sometimes battling other players and virtual characters. Players begin the game with simple and low-powered ships and improve their vessel over time while they complete missions.

The game takes place on a pre-determined area, and the ships correspond to the player devices. Players have to walk physically in the game area to "sail around" and visit places.

The player's equipment corresponds to PDAs fitted with WLAN cards and proximity sensors. The game implements a client-server architecture, where the clients access the server through a WLAN.

The game area (named "game arena") is a small physical place augmented with short-range radio beacons. Those beacons represent game elements like the virtual islands. The game uses proximity sensors in the client devices to detect the islands (beacons) and other players. Hence, the players have to get closer to the beacons to visit islands. If a player wants to battle another player, he has to get

physically close to this player. Likewise, the player has to run away from other players to avoid getting caught in combat situations.

Another interesting design decision relates to have players focus on the environment and social interaction, rather than their devices, as they describe (Björk *et al.* 2001):

> "All interactions within Pirates! are initiated by players. That is, nothing happens in the game unless a player gives the game input. This input can be choosing one of the buttons on the hand-held device or moving oneself across the area where the radio beacons can detect senders. The decision not to include system-initiated actions in the game was necessary in order to allow players to focus their attention on physically navigating in the real world. It also had the advantage of letting players move smoothly between playing the game and socializing with other people in the game environment."

A.3
Botfighters (2001)

Sensors	:	cell-id
Network	:	operator (SMS)
Support staff	:	-
Others	:	pioneer commercial pervasive game

In *Botfighters* (Sotamaa 2002) players control robots and their mission is to locate and destroy other robots (players). The game offers a web module where players are able to customize their robot. The mobile phone is the interface for searching and battling other players.

The "game arena" takes place on the entire city, potentially. Players have to wander around to interact with other players.

Players interact with the game by sending text messages (SMS) to the game server. There were specific text messages corresponding to command to search for nearby players, shoot a specific player, and others. Players would be warned if they were being hunted by other players.

The game relied on GSM cell-ids to calculate if players were close to each other, and the success of the attacks depended on several variables like player shields and weapon power.

In this game, players did not have to meet physically to play. Players interacted with other through nicknames, although they could guess the location of other players based on how frequent they found that player in the same places.

A.4
Can you see me now? (2001)

Sensors	:	GPS
Network	:	WiFi
Support staff	:	yes
Others	:	event game, mobile and web modules

Can you see me now? (Benford *et al.* 2006a) is a research project by the Mixed Reality lab at the University of Nottingham and the artist group Blast Theory. This is game of hide-and-seek, but not as the traditional one. Here there are two teams (hiders and seekers), where one team (seekers) plays in the physical world and the other team (hiders) plays in the virtual (online) world.

The seekers (runners) play on the streets with PDAs accessing WLANs and using GPS receivers. Their job is to chase online players.

The online players navigate through a 3D model of the city, and are able to interact with other online players (through text messages) and see the location of street players represented on the city model.

The street players are able to see the position of their teammates and online players represented on the PDA screen, and also access online player communication. They also have walkie-talkies to communicate with other street players. The game streams this conversation to online players, which makes it possible for online players to have a perspective of what is happening on the streets.

The street players actually were performers from Blast Theory, while the online players were end-users. In order to keep the games happening, it was necessary to perform orchestration. There was a supporting team on the backstage helping the street players to better shape the experience of online players. This process was necessary to deal mainly with issues related to technology limitations and to keep the mixed-reality consistent.

Another interesting aspect of this project is that it could also be characterized as an "event-game".

A.5
Mogi (2003)

Sensors	:	cell-id, GPS
Network	:	operator
Support staff	:	-
Others	:	mobile and web modules

Mogi (Joffe 2007) was massively multi-player mobile role-playing game developed by the French company Newt Games. *Mogi* was deployed commercially in Japan, 2003.

The game corresponds to a treasure hunt where the main goal is to collect virtual items. Those items are laid over a map representation of Tokyo. The players have to roam around the streets of Tokyo to collect those items located at the correspondent map locations. A player is able to capture an item if he gets close around 400m of the item. The game has two modules, a mobile one (runs on the phone) and a web module.

The web module displays a map of the city and also the positions of collectable items. Players using the web module are able to communicate with mobile players through text messages. So, this opens up possibilities to create collaborative activities for the players using different kind of devices.

Players can meet physically to exchange items to grow their collections. The goal is to build up collection and amass the largest possible score.

There is not many works explaining the inner works of *Mogi*. However, this is an important example of pervasive game deployed commercially. It seems that Mogi uses a combination of GPS and cell-id to locate the items (Joffe 2005), where cell-id can be used to make gross assumptions about localization where GPS can be used to fine-tune the results. Mobile players also used the operator data services to connect to the game server and to other users.

A.6
Uncle Roy all around you (2003)

Sensors	:	- (self-reported, manual positioning system)
Network	:	WiFi
Support staff	:	yes
Others	:	mobile and web modules

In *Uncle Roy all around you* (Benford *et al.* 2004a) , players collaborate in order to find the mysterious character Uncle Roy that is hiding somewhere in the city. This game was developed by the Mixed Reality lab from the University of Nottingham and the artist group Blast Theory.

Similar to *Can you see me now?*, the game divides players into two groups: street players and online players. The online players navigate a virtual model of the city, where they are able to explore it in many ways and perspectives, maybe helping street players, maybe not. The street players start the game by heading to the game headquarters. There, they receive a PDA that they use to play the game, leave their belongings, and receive instructions on how to play. Now they have one hour to find Uncle Roy's office in the city. The game is designed around the social theme of "trust in strangers" (remote players, passers-by, anyone), and this reflects for street players as they go around the city with only the PDA and at the mercy of (unknown) online players and the mysterious Uncle Roy.

As street players progress in the game, they are instructed to go to certain places in the city. When they arrive there, they receive clues and messages from Uncle Roy. The PDA displays a fraction of the city map, and the players have to follow it and keeping on dragging an "I'm here" icon over it as they walk on the streets. The game uses this system as a way for players to provide positioning information – the game uses a manual positioning system ("self-reported"). The game designers divided the map intro regions of interest, and whey the players interact with the map, they "know" that the player is likely to be inside that region. In this regard, the game uses a "low-tech" positioning system, but it must trust players. However, the messages delivered to players from Uncle Roy references local context so that they make sense only if players are at that place.

The game supports 12 street players and 20 online players simultaneously. There is also a support team on the backstage to handle issues that might arise during the game event.

The online players can interact with street players by receiving short audio messages from them and sending text messages in return.

Uncle Roy's office is a standard apartment located in a building somewhere in the city, decorated as a game scenario to support the game narrative. When street players arrive at the building, they ring the buzzer and are allowed to enter the office (this is automatic, they do not interact with people in this activity). Once there, online players are able to join the experience: a live video stream is broadcasted online through a web cam. This is the first time online players see the street player. The street player is asked to write a postcard, and in the meanwhile the game asks the online player if they would be willing to support a stranger that was having a personal crisis anytime over the next 12 months.

After the street player finishes writing the postcard, the game asks the street player to go outside and reach a telephone cabin. The player receives a phone call instructing him/her to go to the other side of the street where a limousine stands, and to get into it. Once inside, a man (an actor) asks the player the same questions the online player has answered. The limousine drops the player near the headquarters, and the postcard is dropped into a game box.

Later, if the street player agrees (in the question), they are paired with a random online player who also has agreed, and personal details are exchanged between them. What happens next as a consequence of this is up to the players.

After that moment, the online player is free to continue exploring the city and helping other players.

Some noteworthy features of this game are their nature as "event-games". Also, the gameplay unfolds in a way for street players (based on Uncle Roy's clues) that cannot be sure at some moments of who is playing and who is not. The game transforms the physical place into a mystery environment reinforced by using physical objects (places, cars, phone calls) and actors as live non-player characters, contributing to make the mixed-reality connection stronger.

A.7
The Journey (2004)

Sensors	:	cell-id
Network	:	-
Support staff	:	-
Others	:	commercial (free)

The Journey is a game available for Symbian mobile phones. Andreas Jakl (2004) describes the game:

> "The Journey is a location–based adventure game. The user plays the role of a detect-ive that has to solve a case, commissioned by a mysterious man. The player has to move in the real world with the mobile phone to continue the story and to progress in the game.
>
> The story starts in the bureau of the detective. The next part plays out a bar. To continue, the player has to change his location. The phone tracks the movement using cell–ids of the GSM network and continues the story once the player has reached a new location. Those ids are also stored so that the game remembers locations where the player has already been before. That functionality is being used in the game, which requires the user to return to the place where the whole story has started to participate in the showdown.
>
> How far the player has progressed in the story is automatically saved; the player can resume the game if one is active. The story is visualized using text and a picture, which either shows a scene related to the story, or the general location picture."

The game is highly adaptable to many places, as it structures the story using relative positioning. An interesting thing about the game is the way it handles uncertainty, by providing guidance and information ambiguously, as for example: "continue searching for the bar. It has to be close to your current location".

A.8
Songs of north (2004)

Sensors	:	cell-id
Network	:	operator
Support staff	:	-
Others	:	-

The game description by the authors (Ekman *et al.* 2005):

"The song of north is a multiplayer enhanced reality game. The game draws on inspiration from the Finnish mythology, especially the epic Kalevala. The background story revolves around the legendary Sampo, a machine that is able to produce all the riches of the world. Sampo has been destroyed in the battles between the Northmen and the sons of Kaleva, and now its pieces have become scattered all around the world. The pieces of Sampo are powerful magic elements, but they also cause energy to flow away from the earth. The players, the few Shaman heroes left in this world, now have two options: As they find the pieces of Sampo, they either return them to the earth by dropping them into swamps. With the magic of Sampo returned to the earth, the world will become a better place and nature will flourish again. However, a shaman can also choose to keep the pieces of Sampo and use the magic to gain personal power (at the expense of the state of the world).

In addition to either destroying or keeping and protecting pieces of Sampo, shamans can also find other things to do in the game. The elder gods of the Finnish pantheon have left different quests to be performed by the heroes of the spirit world. These quests may be about finding and combining items in order to gain more power, rescuing non-player characters out of peril, or killing evil monsters. Some of these quests cannot be performed alone, and require co-operation with other shamans. The spirit world is manipulated by casting spells. The shamans cast spells by drumming a virtual shaman drum, which is implemented in the mobile interface.

Players can also interact with each other as well as with the many non-player characters (NPCs) of the world through their mobile devices. The interaction can take the form of fights or collaboration. The game also supports messaging between players.

The spirit world exists in parallel with the physical world. A schematic overview of the game world is presented in Figure 2. Some places in the real world have significance to the game. For example, a portal to the underworld may be situated in a graveyard, water spirits inhabit areas near lakeshores, and so forth.

To be able to interact with an item or character in the spirit world, the player has to be within a spell's reach of the object of his/her interaction. To get to the right position in the spirit world, players move around in the physical world.

The movements of players are detected by means of GSM cell positioning and transformed into spirit-world movement [11]. Game data and information regarding each player's actions is communicated via HTTP-requests between the client and server."

A.9
Ere be Dragons (2005)

Sensors	:	heart rate, GPS
Network	:	-
Support staff	:	-
Others	:	-

Ere be Dragons (Davis *et al.* 2005) is a game for PDAs that shares a social concern to stimulate people to have healthier habits, through physical activities.

The player uses a PDA, and the game uses heart-rate and GPS sensors as input methods in the game. The heart-rate sensor is attached to players and sends data to the PDA over the air.

As the player go to the streets with the PDA, the game renders a "map" on the device screen, where the colors and format depend on heart-rate data.

When the game starts, it asks the player about his age. The game uses this information to calculate the heart-rate range it uses, mainly the minimum "acceptable" value (for game performance reasons), the maximum (a safe value) and an optimal value that the player should strike to maintain.

The game uses the GPS to keep track of locations, so when the player comes back to a previously visited place, the game displays the new map overwriting the previous one.

If the player does well, the map becomes more colorful and expressive, and if the players does not perform well the map becomes darker. By "doing well", the game understands the player keeping the heart-rate around the optimal value. Walking too slow or too fast make the player "do bad".

The game also provides feedback through a set of music tracks, which vary in rhythm ("slow to fast"). The game switches the current track based on the player heart-rate.

A.10
Tycoon (2005)

Sensors	:	cell-id
Network	:	operator
Support staff	:	-
Others	:	seamful design

Tycoon (Broll and Benford 2005) is a multi-player mobile phone game where the goal is to collect virtual items and climb the high-score list. The game

takes place on the city streets, and it uses GSM network cells as a way to divide the city into virtual locations.

The game locations can be either "mines" or "broker". The mines are sources for the game currency ("virtual gold, silver, and copper") that player uses to acquire items that brokers hold.

The mines have unlimited resources, and players do not compete for them. However, the existing items in brokers are of limited availability. The game defines items of different values and kinds. A player is able to buy an item when he reaches the item location, and holds the item price in virtual currency. Once bought, the players receive points related to that item. Once there are no more available items in the game, the game is over.

The game has a client-server architecture. The client is responsible for telling the player where /she is (the virtual location), helping in player navigation and informing the amount of currency the player holds. The server maintains the global game state (available items, items bought, etc.). The clients can access the global game state by placing a request to the server for his information.

The game designers have identified some possible issues related to technological limitations and availability, mainly network cell location handling, network availability and data service price. In order to handle those issues, they employed a paradigm known as *seamful design* (Section B.10). This design approach recognizes technological limitations and incorporates them as part of the application (instead of trying to fight them).

For example, mobile network cells have varying size and their position fluctuates over time. In order to minimize uncertainties related to mobile network cells, the game does not display a precise map of the locations. Instead, the game mere alerts the players when they enter another "virtual location".

Another example is the cost to use mobile connections (required to access the global game state), which was very expensive at the time (and still is on some places). To handle this issue they designed a game mode that encouraged players to stay offline, where they can collect a greater amount of currency. For example, the game would grant more points if the player buys an item using more valuable "coins" (like gold ones). However, staying offline would raise the chances of get-

ting inconsistencies (like trying to buy an item that has already been taken by another player). This trade-off now becomes part of the game strategy (instead of an annoying game issue).

A.11
Epidemic menace (2006)

Sensors	:	GPS, software sensor (weather)
Network	:	WiFi
Support staff	:	yes
Others	:	cross-media, event game

This is a cross-media game. *In Epidemic Menace* (Lindt *et al.* 2007), players collaborate to stop a villain scientist from spreading a deadly virus into the world. The viruses are out in the real-world, growing and moving according to weather conditions.

Players are divided into teams and have several devices at their disposal, each one with a distinct role in the game. The teammates play on different places and have different roles. There is a control room where part of the teams envisions the game globally through a display to analyze and observe the virus, and communicate with other players. Other team members have to go outside to find and destroy the virus. Those players use other mobile devices: a PDA fitted with a GPS sensor, a mobile phone, and a mobile augmented-reality system. The players use the mobile AR system to see the virus in the real-world. The game uses the PDAs to track the players outdoors. The mobile phone is the "mobile assistant", which players use to scan and capture viruses, and also to communicate with the control room. There is also an audio mobile interface to interact with the viruses, mainly to hear about the positions and other properties of viruses.

A.12
Feeding Yoshi (2006)

Sensors	:	WiFi
Network	:	WiFi

Support staff   :   -

Others   :   seamful design

This is a game for PDAs that uses WiFi access points as sources for content.

The game applies the seamful design approach to handling technology limitations.

Here is the game description by the authors (Broll and Benford 2005):

> "The aim of Feeding Yoshi is for each team of players to collect as many points as possible, by feeding Yoshis the fruits they desire. Yoshis are creatures that players find scattered around the city and which are constantly hungry for five fruits, of seven varieties. In order to collect fruit, players must first collect seeds from the Yoshis themselves—each Yoshi always has a seed for the fruit it most often enjoys. These seeds can then be sown at plantations that can be found scattered around the city, just as Yoshis are. Once a seed is sown, the plantation will begin to generate fruit, which can then be picked and used to feed Yoshis
>
> As a player moves through the city, nearby plantations and Yoshis appear as names in a pull down menu and as icons on a map (Figure 1). An audio alert is also made when a plantation or Yoshi is detected so that the player does not have to continually visually attend to the PDA screen.
>
> The Yoshis and plantations that are detected while playing the game are actually wireless access points. As a player moves around in the city, their PDA continually scans for the presence of wireless networks. Secured wireless networks become Yoshis and open networks become plantations."

A.13
Hitchers (2006)

Sensors   :   cell-id
Network   :   GPRS
Support staff   :   -
Others   :   -

The game description by the authors (Drozd *et al.* 2006)

Hitchers is a location-based game for mobile phones that utilizes GPRS networking and positioning using the cell-IDs from the players' phones to create an experience based around the metaphor of digital hitch-hiking.

In Hitchers, initially the world is empty but as the game is played the streets fill with characters who are trying to hitch-hike their way across the city or up and down the country. They have been created and released into the wild by their owners and are trying to find their way home, reach a specific destination, carry out a mission, or just share a journey with a stranger.

Once created, the player can 'drop off' their new hitcher, releasing it into the world to begin its journey. Metaphorically, the hitcher is now removed from their phone and waits in their current location for other players to come by and give it a ride. Whenever a player drops a hitcher they are prompted with the question "Where are we now?" which encourages them to enter a text label describing their current location in the physical world

… In this way, hitchers make their way from phone to phone, player to player and place to place, trying to reach their destination and gathering answers to their question as they go.

… Hitchers currently has no specific goal, scoring mechanisms or game play beyond creating hitchers, moving them around the world and answering their questions.

A.14
Manhattan story mashup (2006)

Sensors	:	camera
Network	:	operator
Support staff	:	yes
Others	:	event game, mobile and web modules

The game description by the authors (Scheible *et al.* 2007):

Manhattan story mashup (MSM) combines the web, mobile phones and one of the world's largest public displays in Times Square to a large-scale pervasive game.

In Story Mashup, individual keywords of textual stories written by web users are presented, one word at a time, to mobile users for the purpose of taking a matching photo with their camera phone. Each resulting keyword-photo pair is validated by presenting the photo together with the original keyword and three other words to two other mobile users, who are asked to choose the most appropriate word given the photo. If either of the two chooses the original keyword, the photo is approved into the resulting visual story. All resulting stories are displayed on the web and selected best ones on a large public display. The players are awarded points for taking photos and for choosing the original keyword.

… The best stories were shown on the Reuters Sign in Times Square in real-time.

A.15
PAC-LAN (2006)

Sensors	:	RFID

Support staff : yes

Network : operator

Others : event game

PAC-LAN (Rashid *et al.* 2006) is a mobile phone game inspired on the classic *Pac-Man* (Wikipedia 2011c). In this version, the game arena takes place in the Lancaster University (England) campus. The game designers had chosen that area because they claim that it shares similar physical traits to a maze.

The game supports five players. One of them is the main character (PAC-LAN) while the others are the ghosts. As in the original game, the main player has to collect the pills and run away from the ghosts, and the ghosts have to catch the main character.

All players wear clothing fitted with radio-frequency (RFID) tags. The game uses those tags to sense when a player is close, so a player is able "to capture" another player.

The pills correspond to plastic discs fitted with RFID tags. Before the game session begins, game staff spreads the discs over the campus, putting them at key locations. Those discs act as tangible objects. The players have mobile phone equipped with RFID readers, so they can interact with all game objects.

The main character "collects" a game pill by having the mobile phone read the RFID tag attached to the pill. A player "captures" another player when his mobile phone detects the RFID tag attached to the other player clothing.

The game has a client and a server module. The client module runs on mobile phones and is responsible for displaying information about the players. The client communicates with the server through the cellular network. The position of *PAC-LAN* is determined implicitly when the player interacts with a pill. The ghosts interact with the pills to have information about the position of *PAC-LAN*.

A.16
Day of the Figurines (2007)

Sensors : -

Network : operator (SMS)

Support staff   :   yes

Others   :   -

The game description by the authors (Flintham *et al.* 2007b):

"Day of the Figurines can perhaps best be envisaged as a massively multiplayer board game that is played using text messaging on mobile phones. The game follows twenty four hours in the life of a small virtual town. Each player chooses and subsequently controls (via their phone) a small plastic figurine that represents their character, journeys through the town, meets and talks to other figurines, visits destinations, finds and uses objects, resolves dilemmas and undertakes missions.

The twenty four hours of virtual game time are mapped onto twenty four days of real time. Unlike most simulation games in which game time is usually accelerated relative to real-time, in DoF it is slowed down so as to deliberately create a slow game that unfolds in the background of players' ongoing lives, perhaps only involving the exchange of a few text messages each day.

The objective is deliberately ambiguous; players are released into the town and told that their goal is to help other players, the rules that govern the virtual city have to be discovered, and there is a strong emphasis on emergent game play in which players construct elements of the game through the exchange of SMS messages.

The back story to the game is that the players are refugees who have arrived in a British Town. The players have to learn how to survive, get to experience various events within the town and ultimately have to decide whether or not to side with an army of soldiers who enter the town towards the end of the game. Players can become more or less healthy and can even die, but beyond this there is no explicit winning or losing. Rather the game is concerned with exploring and constructing a shared narrative through role play, hopefully resulting in an engaging and even provocative experience. In this sense, it is a blend of artistic performance and computer game.""

A curious fact about this game is that there was a game board (like a city mock up) on a public venue, which was a physical representation of the game (Flintham *et al.* 2007b):

"Another performative aspect of DoF can be found in the use of a physical game board which is housed in a public venue (the National Museum of Singapore in the most recent deployment).

The board is a large and distinctive physical structure which shows the destinations within the town and the positions of the figurines that are playing at any moment in time. Players have to visit the board to register and it is therefore their first point of contact with the game

The board is continually tended by a team of human operators throughout the ten hours of every day when the game is active. These operators register players and manually move physical figurines across the board, following instructions from the game engine, projected onto the table as a series of visible augmentations.

> Operators are therefore publicly performing the operation of the game, revealing its inner workings for new players and passing spectators, serving to attract attention, generate interest, and frame the overall experience for new players."

A.17
Insectopia (2007)

Sensors	:	Bluetooth
Network	:	GPRS
Support staff	:	-
Others	:	-

Insectopia (Peitz *et al.* 2007) is a game for one or two players where the goal is to wander around collecting virtual insects, and then building the most valuable collection. To reach this goal, players compete among themselves or collaborate in pairs for collecting. Another for players is to trade insects among them.

The game uses Bluetooth devices to represent the virtual insects. Any Bluetooth device is eligible as game content source. This could be the player's neighbor, some stranger on the streets, or the office printer. The game assigns uniquely insects to the devices, meaning that a given device is always associated with the same insect.

The insects have a life span, defined in the game as eight days. After that period, the insects die and the players have to find them again. This makes playing the game a constant activity to maintain the insect collection.

The game does not use location information. The players have to wander around sensing Bluetooth devices with their mobile phone. The game also does not provide direct player interaction, which they can achieve through other external media (web sites, emails, social networks, etc.).

The game client and server communicate through the cellular operator network.

A.18
REXplorer (2007)

Sensors	:	GPS, camera, Bluetooth
Network	:	-
Support staff	:	-
Others	:	place configuration, tourist game, tangible objects

The description of the game by their authors (Ballagas *et al.* 2008):

REXplorer is a mobile, pervasive spell-casting game designed for tourists of Regensburg, Germany. The game uses location sensing to create player encounters with spirits (historical figures) that are associated with historical buildings in an urban setting. A novel mobile interaction mechanism of "casting a spell" (making a gesture by waving a mobile phone through the air) allows the player to awaken and communicate with a spirit to continue playing the game. The game is designed to make learning history fun for young (and young at heart) tourists and influence their path through the city.

The game does not require supervision while it happens, but it requires personnel to sell tickets and rent devices to players.

The game device is a mobile phone augmented with a GPS sensor disguised as the game object with a protective shell. This shell also provides a physical interface (a customized keypad) for player interaction.

The game detects the gestures using the mobile phone camera. The external GPS sensor was necessary as at that time mobile phone with internal GPS sensors were very rare.

The developers also have spread Bluetooth beacons across the game locations to improve the performance of the game when dealing with locations

A.19
Mythical: The Mobile Awakening (2008)

Sensors	:	Bluetooth, software sensors
Network	:	operator (internet), SMS
Support staff	:	-
Others	:	-

The game description by the authors (Korhonen *et al.* 2008):

"Mythical: The Mobile Awakening is an asynchronous slow-update multiplayer game where players access a magical world through their mobile phone. The magical world is divided into four factions (Dawn, Sun, Dusk, and Moon). The players gain experience and learn spells by completing rituals either alone or together with other players. The spells are then used in encounters to battle against AI opponents or other players (Figure 1B). The game content is based on folklore mysteries and local history for creating an exciting atmosphere.

The game features context-aware gameplay where the real world phenomena have an effect in the game world. Context information derived from the real world is used in the rituals where the reward of the ritual depends on how well the player has met the context conditions set initially (Figure 1A). There are three types of context information used: spatio-temporal, environmental and proximity. Spatio-temporal context is used in two ways: players select a home base from the predefined list and the game content and some environmental context information is then validated against information on that location. Time of the day is frequently used context information that defines when some rituals can be completed. Environmental context information is based on temperature, cloudiness and astronomy. Temperature is used in a breakpoint manner; some rituals require that the temperature is either above or below 0 degrees Celsius. Cloudiness has three possible options: clear, partly cloudy, and cloudy. Astronomy information is related to the Moon and Sun positions over the horizon and to the phases of the Moon. The proximity context is based on scanning Bluetooth devices. Rituals can require scanning either a specific or a given number of Bluetooth devices."

A.20
GPS Mission (2008)

Sensors	:	GPS
Network	:	?
Support staff	:	-
Others	:	commercial (free),
		mobile+web module, multi-platform

In *GPS Mission* (2008), players access "missions" that mainly consist of going to checkpoints guided by GPS. When players get to checkpoints, they receive "virtual gold" that they use to claim special virtual trophies.

While playing missions, players might have to solve riddles, and also to leave "marks" indicating that they have been to that place.

The game has a web-based mission builder where players can build new missions and share them with other players. The game web-site has a ranking of engaged players.

A.21
Seek n' Spell (2009)

Sensors	:	GPS
Network	:	operator
Support staff	:	-
Others	:	commercial

Seek n' Spell (Retronyms 2009) is a commercial game available for iPhone and Android mobile phones. This game uses the GPS sensor in phones to place virtual letters in the physical world locations. The goal of the game is to gather the virtual letters and form words with them.

There are not many throughout analysis of this game, but it seems this game suffers with placing virtual content (letters) in unreachable places, like inside buildings on an urban city. Section B.10 discusses issues as this.

A.22
Gigaputt (2010)

Sensors	:	GPS
Network	:	operator
Support staff	:	-
Others	:	commercial, tangible object metaphor

Gigaputt is a commercial game for the iPhone. The game definition by the authors (Gigantic Mechanic 2010):

> "Gigaputt is an iPhone app that transforms any cityscape into a virtual golf course. Using the GPS capacity in the iPhone, Gigaputt builds a three-hole golf course situated within a two-block radius of the players' location, set onto a satellite map view of the area. Virtual fire hydrants acting as obstacles for the ball and virtual coins that can be collected for extra points add to both the urban flavor and gamesmanship. The iPhone serves not only as course display and locator but also as the golf club, logging the speed and force of players' swings to calculate the distance their virtual balls travel. The app tracks the players' location, so after teeing off, players walk down the block to where their virtual balls have landed, often tracing and retracing their paths throughout the course of a game. The app will not let a player take their next swing until they are within 75 meters of their ball, granting that leeway in case the ball has landed in a neighbor's fenced yard or inside a closed building. "

A.23
Gbanga Famiglia (2010)

Sensors	:	cell-id, GPS
Network	:	operator
Support staff	:	-
Others	:	commercial, multi-platform

Gbanga Famiglia [http://gbanga.com] is a mixture of pervasive game for mobile phones and social networking. In this game, players can start *Mafia Famiglias* or join an existing one. The main goal is to take-over establishments from rival *Famiglias*. Those establishments are virtual content placed on the real-world. More about the game by the authors (Gbanga 2010):

> Gbanga is a virtual world, which is connected to real-world locations you visit. Access to Gbanga's virtual world is via your mobile phone, which you carry around whilst going about your daily life. When your physical location changes, so does the virtual world of Gbanga. By walking around your city you, likewise, move between areas of Gbanga, exploring new places, discovering cool items, meeting other players and... Gbangoos.
>
> What can I win? You score points for completing quests and collecting special items. Rewards range from worldwide leaderboard rankings to sponsored contests with real prizes.
>
> Gbanga is a global community of friends and players connected through location-based quests. At any time of day, anywhere in the world, mixed-reality adventures are awaiting nearby. The platform also features a free alternative to mobile messages (SMS).

The game has different clients for each supported platform (iPhone, Java ME, Windows Mobile, Android, and Blackberry).

A.24
GEO Hunters (2011)

Sensors	:	GPS (?)
Network	:	operator
Support staff	:	-
Others	:	commercial

GEO Hunters is a commercial game for the iPhone. Game description by the authors (YD Online 2011):

> "GEO Hunters is a location-based role playing game that lets users defend their neighborhood and, ultimately, the world from monsters, while competing against other players for top rank.
>
> Set against Maps backdrops, monsters have invaded the Earth, threatening your neighborhood and the planet itself. But don't despair; you're a GEO Hunter who can launch bombs at the monsters and capture them. Once a monster is defeated, you own that monster but must feed and strengthen it or risk losing the creature to other hunters. The more monsters you own, the higher your rank in the game."

Appendix B
Detailing pervasive game features and checklists

This appendix provides more details on the pervasive game features, including the verification questions (checklists) for each item. All games referenced in this appendix are described in Appendix A.

B.1
Local space redefinition

This feature relates with how the game is able to change the meaning of the places where the game are played. By changing meaning, we denote augmenting value to places, incorporating the place (or objects belonging to the place) as game objects, integrating live (human) non-player characters, or making the players perceive the place with alternative viewpoints. This is different from just "being on a place", or using some property (like location) without referring to the local context.

Related features: Mobility, Game object tangibility, Involving non-players, Connectivity, Uncertainty handling policy.

B.1.1
Motivations

When researchers stress that pervasive games are "games coming back to the real-world", they often refer to the physical setting where the game happens, in many cases this corresponds to public areas of urban places. The place is referred to as "the game arena" – a concept that has represents local space augmented by properties from the game. For example, the *Gigaputt* game is a pervasive game based on mini-golf. The golf course stretches over the city, and players have to move physically to different places to follow their shots and get to the holes. In real golf, the equipment to hit the balls is a club. *Gigaputt* represents the clubs with the mobile phone itself, and players have to swing the phone like they would

do with a real club to hit a virtual ball in the game. Figure B.1 illustrates a player playing *Gigaputt[100]*.

Figure B.1: Player playing *Gigaputt*

There are several game objects that inhabit the game world, like manholes, fire hydrants, coins, etc. In this regard, this game transforms the local space into a mini-golf course with virtual content embedded into the mixed-reality.

Another example is *Insectopia* (see Section A.17). The game generates the insects based on Bluetooth devices. Due to the Bluetooth feature of devices having unique identifiers, the game is able to associate insects with specific devices. This opens up the possibility of players associating real-world objects to game content, helping them see the local space (or their elements) in a different way. In this case, the local space becomes more integrated to the game experience as perceived by the player. Peitz and co-authors (2007) even stated that during *Insectopia's* research, a player "specifically went to a *familiar* library to play this game just because she knew that there were many laptops in there".

Sotamaa (2002) refers to those aspects as "social construction of space" the game provides.

Waern and co-authors (2009) present the notion of "authentic environments", which allows "allows the participants to physically act in a near-perfect representation   of the game world", being one of the elements that bring

100Figure used in a basis of fair use policy. Source:
   http://www.flickr.com/photos/giganticmechanic/4254272935/sizes/m/in/photostream.

immersion to a pervasive game. In those environments, "places and objects represent their in-game counterparts: a house in physical world represents an identical house in the game world" (Waern *et al.* 2009). This idea relates to the one we are presenting in this section.

B.1.1.1 Increasing awareness of the environment

Another example is the game *Feeding Yoshi*. The game uses WiFi access points properties as sources of game content. The main gameplay consists of finding plantations (source of fruits) spread across environment and them feeding the Yoshis, which are non-player characters. Secured WiFi access points become the Yoshis and open WiFi access points become the plantations.

During the game evaluation, Bell and co-authors (2006) have received feedback from players that were used to play the game during their daily course to work. Then they had started experimenting taking different paths to "explore" the game, and in the end they had changed their daily course just because there was "good stuff" in the alternative routes they had tried.

B.1.1.2 Reshaping places

This means incorporating elements of local space into the game, as game objects. For example, the game might use some significant landmarks as explicit reference points, or associate some game elements to place features implicitly. The first form would be something like, "go to the square where the statue of King George resides ...". The game *Mythical: The Mobile Awakening* is an example for the second form. In that game, there is a "spirit world" that co-exists with the real-world, forming a mixed-reality. The players are able to interact in the spirit world in several ways. However, there are physical places that have subtle meanings for the game. For example, a real graveyard might work as a portal to the underworld, and "water spirits" might manifest themselves in the game when the player is near lakes or rivers.

From a technology point of view, those approaches have several advantages, as:

- The landmark possibly is a "big" object (area-wise) and has a known (fixed) location in the world. This may minimize problems with errors related to location systems, being an attempt to match the precision of the technology to the game activity;

- Game designers may use ambiguity to refer to the landmark ("go near the statue ..."), meaning that the game can embody technology imprecision in the gameplay, instead of referring to a precise location;

- It is a way to reframe the local space, possibly improving the player immersion with the game, adding up a "fantasy factor" to the gameplay.

B.1.1.3 Using live non-player characters and physical objects

Another approach to convey the sensation of "being on a parallel mixed-reality" is to use live non-player characters and involving non-players (Section B.12). This is noticeable in games like *Uncle Roy all around you* and others by their authors.

Using live characters means incorporating professional actors in the game, interacting with players in the local space. This is something Benford and co-authors (2004a) recommends. McGonical (2006) also shares a similar view when she defends that players engage on a "performance of belief" on pervasive games. We believe this can be relevant due to our experience with amateur theater.

Using this kind of resource characterizes many pervasive games as "event games" or "performance events".

Uncle Roy all around you also provides another example of incorporating physical objects into the game. In the final game stage, players have to reach a public telephone and wait for a phone call. The phone call tells them to wait for a limousine outside, and enter it. Once inside, they interact with a non-player character (an actor), and then the game ends as they are brought to the game headquarters. These aspects are discussed in Section B.2 (Game object tangibility).

Although not a game, another interesting example of integrating physical objects to change the local space is the *World's deepest bin* from the Fun Theory project (Volkswagen 2009). The once ordinary trash bin was augmented with sensors so it would react to people throwing objects into it, play playing a sound resembling a "something falling off a cliff".

B.1.2
Checklist questions

- Does the game integrate physical places (or their elements) in the gameplay?
- Does the game give roles to physical places (or their elements) in the gameplay?
- Does the game help players to have alternative views of the physical environment?
- What physical resources does the game use as game elements?
- Does the game use "live characters" to reinforce the mixed-reality overlay?
- How do game activities affect the physical world?

B.2
Game object tangibility

This feature relates to how the game uses the mobile phone (and other environment elements) as tangible objects, instead of being mere terminals. As tangible objects, we mean that the device has a purpose, role in the game, and the players can manipulate them as game objects instead of terminals (*e.g.* "only "phones). The device could also be disguised as another object through shells and other resources, but the applicability of this approach should be analyzed as it could decrease game accessibility.

Related features: Usability, Social communication, Local space redefinition, Uncertainty handling policy.

B.2.1
Motivations

Tangible object (Ishii and Ullmer 1997) is a metaphor for "human-machine interfaces that couples digital information to physical objects". For example, on a general context, this could be pieces of augmented table-top games (Magerkurth *et al.* 2005). Existing research suggests that this feature contributes to increasing player immersion in the game (Waern *et al.* 2009; Tuulos *et al.* 2007; Lindt *et al.* 2005).

Our approach relates to using the mobile phone associated with sensors, either in the phone or in the environment, to create tangible objects.

B.2.1.1 Mobile phone as physical game object

There are several examples of approaches to incorporate mobile phones and other devices as tangible objects.

Our prototype *The Audio Flashlight* transforms the mobile phone into an "audio flashlight" that the player uses to find a treasure hidden in the virtual world. Players interacted with the audio flashlight through gestures. There where specific gestures to walk in the virtual environment, and the device reacted by playing music (as a radar to the treasure) and vibrations to indicate obstacles. The game uses the phone accelerometer to detect the gestures. In our research, we have verified that players enjoyed this form of interaction.

Pervasive Clue augments a PDA with a physical magnifying glass and RF readers, transforming it into the "cluefinder", which has a significant role in the game.

Gigaputt transforms the mobile phone into a golf club, and the players interact with the phone as they would do with a real club.

REXplorer goes beyond and physically transforms the mobile phone into another device. They have equipped a mobile phone with a GPS receiver (built-in GPS sensor for phones were not available at that time) and a shell that also changed the physical interface, providing another keypad with less buttons and icons fitted for the game's purpose. Players interacted with the device through ges-

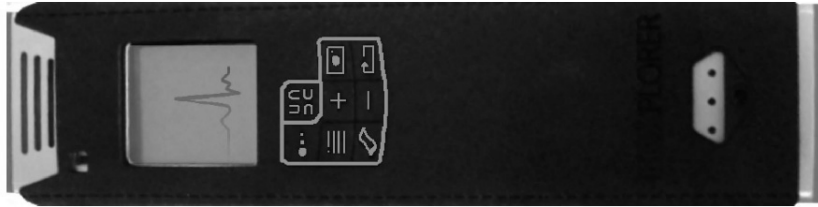tures, which the game detected using the built-in camera. Figure B.2 illustrates this device[101].

Figure B.2: The *REXplorer* "detector"

Tuulos and co-authors (2007) write interesting passages of player behavior due to using the mobile phone as a game object:

> We were surprised to see how effortlessly the players were able to cross the boundaries of ordinary public behavior, *e.g.* by acting out publicly, once they started to look at the world through the game client, …
>
> … Clearly, the use of mobile device as an interaction device and as an image capturing device in the context of our game is strong: "When I was holding the phone, I felt confident hunting for images and doing the guessing part" (player testimonial)

Lindt and co-authors (2005) argue that using tangible game objects improves the quality of interaction in the game.

B.2.1.2 Using multiple props and social communication

Researchers suggest that using multiple props (physical game objects) in the game might improve social communication, immersion, and the game experience (Waern *et al.* 2009; Ballagas *et al.* 2008).

A game prop is a game object designed with a specific role in the game world. In our prototype *The Audio Flashlight 2*, there are two roles players may choose. The roles have different responsibilities, and players have to collaborate to reach the goal, that is finding a treasure in the room while avoiding stepping on bombs that lie in the virtual world. The first role is the "flashlight", the same as the first version of the game (previous section). The second one is the "miner", responsible to detecting mines and telling that to the first player (who cannot detect

---

101 Figure extracted from (Ballagas *et al.* 2008) in a basis of fair use policy.

them). Another prototypes we have implemented (*The Audio Flashlight 3* and *5*) also apply this idea.

The *REXplorer* game also uses multiple props, and an interesting result reported by them was that "players tended to choose one of roles" and "switched roles regularly", suggesting that this resource promoted communication among the players, and they were able to enjoy the different aspects of the game experience (Ballagas *et al.* 2008)

B.2.2
Checklist questions

- Does the game specify a role for mobile phones in the game, or are they mere access terminals?
- How does the game transform mobile phones into game objects?
  Here we are interested in knowing which resources the game applies to make the mobile phone into a tangible game object. For example, the game could use inertial sensors or the camera to detect gestures (as *The Audio Flashlight* series), or disguise the device into another one (as *REXplorer*).

- Does the game use other physical objects (other than mobile phones) equipped with sensors or actuators as game objects?

Here we are interested in knowing if the game uses other physical objects with significant roles in the game.

- Does the game use mobile phones as multiple props?
- How does using tangible objects improve player socializing in the game?

B.3
Game pacing

This feature relates to how technology influences or limits the pacing of the game. Related features: Mobility, Daily life interleaving, Connectivity.

**B.3.1
Motivations**

There are some characteristics of technology that should be taken into account when designing the game experience. For example, some technologies do not work well if the player is moving above some threshold speed, or some technologies may not provide responses in a real-time fashion that might be required by some games. In those cases, it is necessary to design game activities that are compatible with these technology characteristics. This mainly concerns networking technologies and sensors used to detect near objects.

Networking technologies seems to be especially susceptible to this issue. The game *Feeding Yoshi* uses WiFi access points as sources of game content. Bell and co-authors (2006) report that players that were moving fast (on cars, trains) were able to find content (access points) but had trouble connecting to them. On other occasions, players on cars would slow down to be able to catch the virtual content.

Another example comes from *Hitchers* (Drozd *et al.* 2006), which uses location through phone cell-IDs. In this game, they drop or pick virtual characters (the Hitchers) from special locations defined by the cellular network cells. Drozd and co-authors report an example of a player playing the game on a train, and while he passes through a train station, he drops a Hitcher. When passing again through the same station, coming from the opposite direction, the player would expect to pick the Hitcher again. However, in practice this is not what has happened, as the Hitcher would be some distance away from the station (considering the direction of movement). They have suggested that this happens due to "cell elasticity when playing at speed", an effect that is related with the nature of cellular networks (like overlapping of cells) and quick movement. For example the station might be in the middle of two cells. On the first trip, the player is connected to cell 1, drops the character, and later connects to cell 2. The character then is associated with cell 1. Upon return, the player is connected to cell 2 when he passes through the station, not finding the character, which appears later when the player connects to cell 1.

Another examples of problems related to response times is GPS and Bluetooth. GPS sensors can take a long time (like several minutes) to get the first satellite connection, and be available to the user. This also happens when the sensor has to re-acquire the connection. Benford and co-authors (2006a) have reported this for *Can you see me now?*. This is also common for other games.

Some games use Bluetooth as a source for game content. However, searching for Bluetooth devices takes a long time, considering timing conditions of some game activities. In this regard, game activities that need immediate responses about which Bluetooth devices are nearby become unfeasible.

Other technologies (like RFID) seem adequate for fast-paced games. During the *PAC-LAN* research, Rashid and co-authors (2006) have concluded game activities relying on RFID sensors are feasible.

Other projects seek deliberately to slow-down the game pacing, which could be a solution to overcome those technology issues. For example, *Day of figurines* uses interaction through SMS messages, as asynchronous gameplay, and slow pacing. The game lasts for 24 hours in game time. In real world time, one day corresponds to one hour of game time, making the game last for 24 days.

The game pacing also affects interleaving the game into daily life, as the example of *Day of figurines*. Another possibility is to design short play-sessions, so that players are able to play on intervals of other activities. The *Insectopia* game is an example of this.

B.3.2
Checklist questions

- Is the technology the games uses compatible with the game pacing?

There are technologies that work better or worse depending on the game pacing. So, it is necessary to certify if the technologies are adequate e how this affects the game experience

- Do game activities adapt their pacing to accommodate technology limitations in this regard?

The activities may have to vary their pacing to adapt to technology limitations.

- Does the pacing of game activities require the player to focus exclusively on the game? For how much time?

Here we are interested in knowing the game pacing makes it possible to interleave game sessions with other non-game activities. See Section B.6 (Daily life interleaving).

B.4
Mobility

This feature relates to aspects of portability and mobile computing – using wireless connections on mobile devices – and also aspects pertaining to the physical size of the game area, which may require players to move.

Related features: Game pacing , Uncertainty handling policy, Connectivity, Conformance to social and physical setting

B.4.1
Motivations

We consider mobility as "the capacity of moving computing services with us, through reducing the size of devices, providing network access while moving, or both" (Lyytinen and Yoo 2002). We are considering as "mobile games" the ones that are "played on the go", "portable games", and games that do not necessarily use network connectivity. However, we consider as a "the highest level of mobility" the situation where a game is portable, has plenty of network resources available (regarding quality of service and availability), and uses those resources in the game activities.

As we are speaking of mobile phones, this can be considered one of the most obvious features. There are some issues that should be outlined, as requirement of movement, networking usage, and player synchronization.

In pervasive games, the requirement of movement in game space varies greatly. Some games are restricted to small areas (as *Pirates!*), other to some controlled parts of the city or place (as *Uncle Roy all around you*, *PAC-LAN*), and other to very big (mostly unrestricted) areas (as *Botfighters*, *Insectopia*, *Mogi, Gigaputt, Gbanga Famiglia, Seek n' Spell, GEO Hunters*).

The scale of the potential game area greatly influences its realization. For example, games with potential big areas benefit greatly from using mobile phones as those devices are able to use networking anywhere there is coverage from the cellular operator. In this regard, developer can rely (theoretically) on an infra-structure that is already available. In terms of context-awareness, those games will need to be adaptable to variable contexts, though (see Section B.5, "Game content adaptability"). For smaller scale areas, the developers might be able to provide a more customized game experience, as in *PAC-LAN*. This also may help developers in handling limitations of technologies, like network availability (see Section B.10, "Uncertainty handling policy").

Also, the technology used may affect the mobility aspect of the game. As an example the developers of *Hitchers* had used the network cell-IDs as sources of game content, and they have reported that this game does not well in "small" places (a campus, a building), because game uses cell-IDs in such a way that requires players to move through long distances ("across a city") for the game experience to be maximized.

The game pacing also might affect mobility, for example, because networking technologies might not well when the player is moving above certain speeds.

Speaking of networking, game designers should not assume constant communication with servers or other devices. Although the mobile phones should be able to connect from anywhere (where there is coverage), availability and reliability of networks in large areas vary. The designers should establish a strategy to handle those issues (see Section B.10, "Uncertainty handling policy").

B.4.1.1 Stimulating players to move

It is also an option to consider stimulating players to move as part of the pervasive game activities. For example:

- Stimulate players to make physical exercises.
- Stimulating tourism activities, by having the players to explore a number of selected places.
- Minimizing technological issues related to location systems (like GPS), making the players travel large areas.

For example, our game prototypes *Pervasive Word Search* and *The Audio Flashlight 4* stimulate players to move to crowded areas. Some rules on those games favor playing the game in places that have lots of Bluetooth devices, so a crowded place is likely to be a better place to play.

B.4.1.2 World-Player synchronization

This property affects pervasive games that rely on mobility and location, relating to how the game keeps the consistency between the physical location of the player, and how the rest of the system (including other players) sees it. The main question is how the game experience is affected when the game is not able to maintain player consistency. While exploring this issue, Benford and co-authors (2006a) recommended designers to explore activities considering "four possible states of the player: connected and tracked, connected but not tracked, tracked but not connected, and neither connected nor tracked". This is an issue that affects the mixed-reality experience and it is important issue to consider it.

B.4.1.3 Checklist questions

- Does the game need to use networking while the players are moving? If yes, how does the game handle networking limitation issues?

Strategies for handling technological issues are the subject of Section B.10, "Uncertainty handling policy".

- Does the game have location-dependent gameplay?

Here we are interested to know if the game uses player location as input, or if the game activities require the players to move.

- What is the order of magnitude of the potential game area?

For example, we are interested in knowing if the game is restricted to small places, or may happen in big areas.

- Does the game happen in a fixed place?

We are interested to know if the game is bound to a specific place, possibly pre-configured for a game session. For more on this, see Section B.7, "Game autonomy".

- How does the game keep players consistent within the game world?

The issue discussed in former section, "World-Player synchronization".

B.5
Game content adaptability

This feature relates to pervasive games that are context-aware – generating the game content from the environment. It approaches the issue of how those games are able to adapt themselves to various places, keeping the same (or expected) functionality regardless of the environment.

Related features: Uncertainty handling policy, Mobility, Game autonomy

B.5.1
Motivations

Games that rely heavily on sensors to generate dynamic content fall into the subject of this section. An example would be games that use proximity sensors to generate content.

Our prototype *Pervasive Word Search* relies on camera, Bluetooth devices, WiFi access points, and light sensors to generate the game content (the letters for the target word). In our case, the game could be played virtually anywhere. Our

another prototype, *The Audio Flashlight 4*, uses nearby Bluetooth mobile phones as means to decide whether to make the game labyrinth easier or harder.

The *Insectopia* game scans for Bluetooth devices nearby to generate the insects the players have to collect. Another game that has an interesting approach is *The Journey*. This game has a storyline that involves going to several places. The places in the story are mapped as network cell-IDs. The game uses imprecise instructions like "now you should go to the bar next to you", and game events happen when the game detects changing of cell-IDs.

B.5.2
Evenly-distribution of content

The biggest advantage of a game having high content adaptability is that the game does not need to be adapted to be played elsewhere. However, the smoothness of the game experience depends also on some technological factors.

One problem of relying on context is that it is necessary to foresee where the players are going to play the game to check if the context adaptation works for all places. Failing to do so may decrease the game accessibility, because some players might not be able to play the game (or to play the game properly).

For example, the game *The Journey* generates its story based on the network cell-ids. The game is able to work well in urban places where network cell density is likely to be high. However, when moving to rural areas, this might not be the case.

The same is true for games that use WiFi access points as sources of context. On business areas, the density of WiFi networks is likely to be high, while it might be much lower on residential areas. So games that rely on WiFi for context must take this into consideration.

Also, as Korhonen and co-authors (2008) have pointed out, some context is dynamic and developers have no control of them (like weather). Hence, on some situations using context like this may lead to situations where the players are not able to play. Similar situations are for games that rely on certain dynamic contexts like time of the day ("morning", "evening", etc.), as players might not be able to

play the game under that conditions due to their other activities (jobs, studies, etc.) (Korhonen *et al.* 2008)

Another subtle situation related to the previous one regards global games that have specific events happening at some specific time. For example, the game *Day of Figurines* (Flintham *et al.* 2007a) had two of such events, the "opening hour" and the "closing hour". The game was playable during those hours. However, the hours were determined by a specific time-zone. Flintham and co-authors have reported that many players from distant time-zones had become frustrated as they were not able to synchronize their available time with the game's time-zone.

B.5.3
Checklist questions

- How does technology availability affect the game content and the player experience?
- Does context usage affect mobility?

We are interested to know if the context usage requires players either to move more or move less in physical places.

- Does context usage affect accessibility?

We are interested to know if the context usage either facilitates or hinders players from accessing the game.

- Are players able to play the game in various places without manual intervention/re-installation/re-adaptation/orchestration?

We are interested to know if the game is able to adapt automatically to other places, or are bound to a specific place or features in that place. Sections B.1 ("Local space redefinition") and B.7 ("Game autonomy") discuss pervasive games that depend on specific places or local features.

B.6
Daily life interleaving

This feature relates to how the pervasive game is able to become diffuse through daily life, enabling the player to integrate gaming sessions with other non-game activities.

Related features: Game pacing, Connectivity, Persistency, Uncertainty handling policy

B.6.1
Motivations

This feature relates to the temporality of the game. In the ultimate state of pervasiveness, the game would be available anytime, existing as an "alternate world" that is available all the time, parallel to "real life". The most prominent aspect of this is the possibility to engage with a process that evolves continuously over time (and does not depend whether the player is connected to the game or not).

Daily life interleaving relates to this as it refers to ways that games would be mixed with other non-game activities of players. In fact, creating a pervasive game with this feature requires designing around a persistent game world.

Many traditional games for mobile phones are "casual games", meaning that they have simple gameplay are designed to be played on "intervals of daily life". Although playing those games also interleave with daily life on some degree, traditional (casual) mobile phone games do not persist between game sessions. We include this behavior as relevant for the *Daily life interleaving* feature, but we are more interested in pervasive games that persist over time, as far as this subject is regarded.

B.6.1.1 Interruptions

Interleaving the game with daily life implies that the game will be interrupted sometime. Several authors propose approaches to blending game activities with other non-game ones. For example, (Saarenpää *et al.* 2009; Flintham *et al.*

2007b) propose implementing "asynchronous gameplay", or "asynchronous slow update gameplay". As Saarenpää and co-authors define, this "means that players are able to interact with the game system when an opportunity presents itself, but other players will not notice the differences in actions". Games that apply this approach may have slow pacing, to helping in creating the opportunities for playing. Examples of games like this are *Day of figurines* and *Mythical: The Mobile Awakening*.

As Saarenpää and co-authors (2009) remark, changes in player context may create opportunities for playing, or may cause playing to be interrupted abruptly, an issue that should be considered when designing activities. For example, players are on train and have to get off, or players are waiting on the line and their turn has come.

Examples of potential interruptible situations are when the player is moving, or when he is on a social setting with specific social conventions (hospitals, public places, etc.).

Another concept related to interruptions is "short play sessions", where players are able to join the game anytime, engage in some quick game activities, and then leave the game. An example of game featuring this is *Insectopia.*

In case of mobile phones, traditional or pervasive mobile phone games must consider the device was conceived primarily for voice and message communication. Hence, receiving phone calls or messages are high-priority events for mobile phones. In this regard, game activities must be designed to be interruptible, and they must not interfere in the other non-game activity.

B.6.1.2 Game and player communication

Considering pervasive games that are persistent and exists as an independent process, some events might occur in the game world that might affect players that are not currently playing. An implication of this is that the game should provide a way for players to be aware of such events that might affect them, or require action.

Korhonen and co-authors (2008) point out that the game may use direct or indirect methods to communicate asynchronously with the players. Mobile phones are well-suited for direct notification, being it through SMS or MMS. Email is also an option, but requires the user to access the internet, which may incur in extra costs for the player. The first option leaves the cost to the game hosts.

The indirect notification method would be through web-sites, emails, or other media. In this case, players would have to access this media to check the information, which would make them come aware of important events only (well) after they have occurred. For players, this could generate costs, like internet access.

Another important consideration regarding notification is to choose the granularity of messages delivered to players. Too much messaging might irritable players, as being regarded as spamming.

B.6.2
Checklist questions

- Does the game provide a persistent game world?
- Are game activities designed to be interruptible?
- Do game activities require long time commitment from players?
- Which approaches or techniques does the game apply to blend with daily life?
- Does the game provide equal opportunities for playing in any time?

We are interested to know if players are able to join the game anytime and enjoy the "same experiences", or if game activities rely on some specific time schedule that may hinder players from fully enjoying the game.

- How does the game communicate events to players?

We are interested to know about the notification method, like "direct", "indirect", and how it is implemented.

B.7
Game autonomy

This feature relates to the game as being an independent system that does not require preparation for game sessions (like configuring the physical location) or in-game management by a support team.

A pervasive game that requires preparation and/or in-game management happens at specific times, with a specific duration. This kind of pervasive game could also be considered as "events".

Related features: Local space redefinition, Game object tangibility, Involving non-players, Mobility, Uncertainty handling policy.

B.7.1
Motivations

Event games are pervasive games that happen at specific time and place, with a fixed duration. Those games share some of the following characteristics:

- References to local physical context (specific addresses, statues, buildings, etc.) are part of the game.
- The game area is constrained.
- The place is augmented with technology (as sensors) that enables important gameplay features. This step ("place configuration") happens before game sessions.
- Uses actors as live non-player characters.
- Requires a support staff to supervise the game as it happens, working to keep player experience as smooth as possible, possibly intervening in the game to handle issues so players do not notice them.
- Requires specific hardware (sensors, playing devices, etc.).
- Requires wireless networking

There are several pervasive games that could be considered as "event game events", like: *Can you see me now?*, *Pirates!*, *Uncle Roy all around you*, *REXplorer*, *Epidemic Menace*, *PAC-LAN*, *Manhattan story mashup*.

Event games are a way to provide a more customized game experience to players, as designers are able to control many aspects. It is not uncommon that those games be referred as "performance", especially when it involves actors (Benford *et al.* 2006b). Another indicative of this is authors describing the realization of the game as "staging experiences", or "Since 1998 we have staged three such performances" (Benford *et al.* 2006a; Benford *et al.* 2004a; Montola *et al.* 2009).

Event games can also be an alternative to minimize issues related to technology limitations (which is the subject of Chapter B.10). For example, designers may choose a place where they can provide reliable wireless networking, or choose a place where location technologies work well.

A prominent characteristic of event games is that they often require some level of management while the game is happening. The literature refers to this aspect as "game orchestration". This involves creating custom software tools to monitor game aspects (player status, connectivity, etc.), assembling support teams that work behind the curtains to keep problems from breaking the game experience, sending support staff to streets to help players in trouble, preventing players from getting harm, etc. For example, in *Manhattan story mashup*, players generated content (in form of words and pictures) that were later transformed and displayed in a large public display in Times Square, New York. The game had moderators keeping an eye on player content to filter what they thought would be inappropriate for the game (as "offensive words").

In summary, event games offer the possibility to deliver really unique experiences to players, as game designers are able to optimize the game experience in many ways, by constraining the scope of the game to a "controllable scope". In those cases it is also possible to use custom (an expensive) technology, freeing the game designers to think about more ambitious experiences. Also, using actors might help in increasing immersion. One drawback of such games is the cost associated to realize the game vision, along with management and tools, which might be high.

B.7.2
Checklist questions

- Is the game bound to specific places or local context?
- Does the game require configuring the physical space for a game session?
- Does the game require any kind of supervision when players are playing it?
- Does the game require custom hardware, actors, or other kind of related resources?

B.8
Device independence

This feature relates to how the game is available for multiple mobile platforms.

Related features: -

B.8.1
Motivations

With this feature we are interested to know if players are able to play equally on many kinds of devices. In case of mobile phones, a possibility is to create thin clients that run on the devices, leaving a significant part of the game to live on the cloud. This is the approach games as *Mogi* and *Insectopia* take.

Another possibility is to have one client for each platform, synchronizing them on the cloud. This is the approach *Gbanga Famiglia* takes.

For example, the former approach could be implemented through software based on the Java language, due to the high availability of phones with the Java platform. In practice, this does not work well due to device fragmentation – although the software environment is the same theoretically, the hardware is not, with different screen sizes, display color depths, input methods, CPU processing, and other things.

The other possibility is to develop specific clients for each platform. This is more costly, but it is more powerful as developers are able to tailor the game client to each hardware platform.

B.8.2
Checklist questions

- Is it possible to play the game on multiple mobile device platforms?

B.9
Cross-mediality

Cross-media games that are played with different devices or media, but each device has a distinct role in the game. This is different from a game being available on multi-platforms.

Related features: Game object tangibility, Game pacing, Mobility, Social communication,

B.9.1
Motivation

Cross-media games are games that spread across different devices and media. However, those games are not "multi-platform", that means the game runs on many devices the same way (or provides very similar service). In cross-media games, the devices have distinct roles in the gameplay. Lindt and co-authors (2007) define cross-media games as "games that uses a variety of gaming devices and gaming interfaces that support different forms of participation and deliver different game experiences".

Usually, those games use a combination of stationary and mobile devices. Games as *Can you see me now?*, *Uncle Roy all around you*, *Manhattan story mashup*, *Mogi*, *Botfighers* can be considered as cross-media games. An example of comprehensive cross-media game is *Epidemic Menace*, which had seven different game devices or interfaces to the game.

It is not uncommon that those games use tangible game objects (see Section B.2). Lindt and co-authors (2005) argue that using tangible game objects improves the quality of interaction in the game.

Another important consideration for cross-media games relate to the pace of the game activities when using different devices. For example, game activities for desktop computers (as the web) might have different paces compared to game activities based on mobile phone devices. This occurs because of the nature of the media, or underlying technology.

In a game that uses both classes of devices, it is important to synchronize the pace of game activities so that different devices match. An example from the authors of *Manhattan story mashup* (Tuulos *et al.* 2007):

> A major technical research motivation for having the web site in the first place was to gain better understanding in different time-scales of the web and the physical world, and how it affects the system dynamics.

> In practical terms, minutes make a big difference while one is standing on a busy street in Manhattan, compared to web site which is still mostly conceived as a rather static entity.

> Especially it is not accustomary to have a web site open only for 90 minutes, even though such a happening makes sense in the physical world. Thus to achieve smooth real-time interaction between the web and the physical world, the system must carefully take into account inherent differences between the two time-scales.

Another example by the authors of *REXplorer* (Lindt *et al.* 2007)

> The game state is sent to each of the gaming interfaces at different frequencies, depending on their real-time capabilities, *e.g.*, the smart phone receives game state updates at a lower frequency than the stationary game board.

B.9.2
Checklist questions

- Does it make sense for the gameplay to include multiple devices with different roles? Does the game allow different modes of participation?
- Does the game balance the game experience for the various specialized devices?

If there are devices with "too much importance" in the game (compared to other devices), players that do not use those devices may have a poorer game experience.

- When using devices for multiple platforms (*e.g.* desktop, mobile), do the game activities have compatible paces, so as not to break the gameplay?

B.10
Uncertainty handling policy

Technologies have inherent limitations regarding precision, accuracy, availability and other uncertainties. Hence applications relying on technology have to deal with those issues eventually. This feature relates to how the game handles technology limitation issues.

The original vision of ubiquitous computing by Mark Weiser (1991) is about computing getting into the background so users can focus on their tasks, and not on the machines. This idea seems to have been translated to a vision where technology limitations would be handled seamlessly (or transparently), in the sense that the users would not notice them. A "seam" is a noticeable boundary, break, or gap between technology components. The concept has been extended also to include breaks in the smoothness of user experience, due to technology limitations.

B.10.1
Strategies

Technology limitations arise due to the nature of the different infra-structure components, and it is not possible to eliminate them totally. Users perceive limitations as inconsistencies, ambiguities, or imprecision. Examples of technology limitations include variations of signals, lack of coverage, communication delays, loss of information, limited sensor precision, and environment interference, among others.

Some researchers (Benford *et al.* 2006a; Bell *et al.* 2006) have identified five general strategies for handling technology limitations: remove, hide, manage, reveal, and exploit.

B.10.2
Remove

The *remove* strategy means to design activities so that limitations never appear in the game. This includes using improving technologies or designing activities that fit the technology limitations into them.

Using better technology is an alternative that may not be possible to implement, as limitations might plague the most current version of the technology. This would be long term solution. The second option is the most feasible, in practice.

An example of removing uncertainty comes from the game *PAC-LAN*. The game happens in a pre-determined area, where RFID tags are deployed in the environment. The game determines the location of players when they interact with the RFID tags, thus removing uncertainties related to the location system. Also, the game uses connectivity through the mobile operator, as the author had measured this method of connection and found it good for the game purposes (regarding availability).

Another example comes from *Uncle Roy all around you*, where the game uses a self-reported (manual) system for informing location.

B.10.3
Hide

The *hide* strategy means anticipating issues and correcting them before the player has a chance to face it. We argue that applying "ambiguity" is a good tactic to implement this strategy.

A hiding strategy comes from *Mogi*, where players capture "items" if they are close to a distance of 400m. This definition is precise enough to accommodate the precision of location systems in the game.

In *Botfighters*, players issue commands to find other players, and then the game returns "approximate" information about location, as "player X is in ABC street 400 meters north". The game does not draw maps with the locations, which helps in creating ambiguity.

The game *The Journey* structures the locations in its story according to network cell-IDs around the player. The game interacts with players by suggesting them to reach locations defined fuzzily, as "continue searching for the bar. It has to be close to your current location".

Our prototype game *Pervasive Word Search* defines as "wireless zone" a place with Bluetooth devices or WiFi access points. When entering or leaving those zones, it just informs the player with "you've entered/left a wireless zone" with no further details.

In general, trying not to depict precise representations of data gathered from sensors usually helps in implementing hiding strategies. This includes to trying to display maps about measured properties (like location).

B.10.4
Manage

Managing uncertainties include having fall-backs to use when the primary mode of operation fails. In other words, the game adapts to the circumstances by having several modes of operation.

Examples include designing an offline mode to use in the game when connectivity is not available. Benford and co-authors (2006a) suggest using a manual location system (self-reported) for when the automatic system is unavailable.

Benford and co-authors also notice that in-game management (orchestration) is a way to implement the *manage* strategy. This is the case as games *Can you see me now?*, *Uncle Roy all around you*, and *Manhattan story mashup*.

B.10.5
Reveal

The *reveal* strategy consists of presenting the limitations to users and letting them device how to act. In case of mobile phones, users are already accustomed with the operator signal indicator that those devices display.

Benford and co-authors notice (2006a) that several researchers have been favorable of applications revealing the seams of technology, because of several benefits as: improving user performance when interacting with applications, improving understanding of context-aware systems, providing users more control over the applications.

B.10.6
Exploit

Exploiting limitations means acknowledging their existence and integrating them into the game. This approach is known as *seamful design.* Seamful design for mobile games and other systems has been discussed by several researches, as (Chalmers and Galani 2004; Barkhuus *et al.* 2005; Broll and Benford 2005; Bell *et al.* 2006). In contrast, *seamless* applications regard the limitations as problems that should be avoided. Those applications apply *remove*, *hide*, or *manage* to handle the technology limitations.

Examples of games that apply seamful design are *Tycoon* and *Feeding Yoshi.*

In *Tycoon*, an important seam relates to the mobile network cell coverage. In this game, network cells correspond to game locations. The game handles this issue by assigning roles to the cells, and never presenting a precise map based on them. A cell-based map would be inherently imprecise, due to factors as variation in cell size over time and overlapping cells. When playing the game, the players do not know exactly about the limits of the places (*e.g.* cells), because the game only informs that "locations have changed" (when it detects a change in current network cell). The players do not deal with the notion of "network cells", the game uses a more abstract notion of "game areas/location" as "mines, brokers".

Another seam in *Tycoon* relates to the game connectivity. At first, the game would require players to connect to remote server to update the game state, either to propagate local changes, or to get remote information. This would demand constant connections through the mobile operator, which could generate an amount of traffic that could be expensive for players. At the time the game was developed, operators usually charged data users based on traffic. Also, there was the problem in assuming that connectivity would be widely available and reliable.

They have solved this problem by designing an offline mode, which had several advantages that stimulated players to stay in this state. However, players would need to develop a proper strategy as staying in offline mode might cause them to miss important events.

B.10.7
Mixed-reality overlay

Another issue related to handling technology limitations is maintaining the mixed-reality. In pervasive games the virtual and physical world are mixed, defining a "mixed-reality" where the game world resides. If the mixed-reality is not consistent, the game experience may be decreased. Many authors refer to pervasive games as forming some kind of mixed-reality, for example, (Struppek and Willis 2007; Joffe 2007; Benford *et al.* 2006a). This feature also relates to "immersion".

Sensor technologies ignore physical world boundaries. For example, the game *Seek n' Spell* (Retronyms 2009) is reported to place content inside buildings, becoming unreachable. Sensor signals go through walls and doors, so designers must take this into account.

Another related situation is that sensors might detect another sensor nearby, but players will not be able to interact because there is an obstacle (*e.g.* a wall) between them.

B.10.8
Checklist questions

- How does the game handle technology limitations?
- Does technology correctly acknowledge physical world limits?

B.11
Social communication

This feature relates to how the pervasive game is able to innovate in fostering social communication, acting as a medium for people to communicate. It includes means for communication between co-located people, distributed people, or both. This includes aspects as: community formation, social networks

Related features: Persistency, Connectivity

B.11.1
Motivations

Promoting the social aspect of games is one prominent feature of pervasive games. We discuss in this section some aspects that may help in realizing this feature

B.11.2
Social engagement

The game may use several approaches to foster people to communicate with each other. Some examples:

- The game provides a chat or text messaging facility;
- The game has activities that make players share a common physical location;
- The game has activities that require players to find/talk other players;
- Collaborative activities.

Another example comes from the game *Pirates!*. In that game, players are co-located in the game arena. The game tries to stimulate social interaction by in-

ducing players to engage. When it detects two nearby players, the game offers them an option to initiate a battle.

A more advanced approach would be to provide support for community formation, and social networks around the game. This approach requires the game to support persistency.

B.11.3
Complementary players roles in activities

Another technique to increase social engaging is to design different and complementary roles in the activities, forcing players to communicate among themselves.

For example, each role might provide a unique perspective of the game, making it impossible to play the game using only each perspective.

This design has been applied to our prototype *The Audio Flashlight 2*, where one of the players was a Flashlight and the other was the Miner, with different abilities the players must combine to achieve the game goal (find the virtual treasure in the dark room).

B.11.4
Gameplay emergence

This aspect relates to the game fostering the gameplay to unfold in unpredicted ways, deliberately, or through other means. This also relates to the having a base platform that players may use to customize the experience.

For example, there are games (like *GPS Mission* (Orbster 2008)) that have web-based tools to build game missions or tasks for other players.

Another example is *Insectopia* (Peitz *et al.* 2007), where the game generates content automatically based on nearby Bluetooth devices. In this case, the game does not need custom tools for presenting emergent gameplay.

B.11.5
Checklist questions

- How does the game use technology to provide means to improve communication among people?
- How does the game transform the relationships among players?
- Does the game stimulate players to approach/start interactions with other people?
- Does the game use technology to foster community/social networks forming?
- Does the game present emergent gameplay?

B.12
Involving non-players

This feature relates to integrating non-players into the game. Some games might use this to create ambiguities about who is playing and who is not, creating what Montola and co-authors (2009) have defined as "social expansion".

Non-player participation is not required to be conscious. For example, *Insectopia* (Peitz *et al.* 2007) and our prototype *Pervasive Word Search* use other people's Bluetooth devices as sources of content.

Another way of integrating non-players is using actors to represent live non-player characters. *Uncle Roy all around* (Benford *et al.* 2004a) you uses this.

This idea also relates to players having to discover and meet (physically) who are the other players. By doing so, players might have to approach strangers on public places, and then proceed to complete game activities.

It is important to notice that involving non-players in the game might raise ethical issues.

Event games are particularly suitable for applying this approach.

Related features: Game autonomy, Conformance to physical and social setting, Mobility

B.12.1
Checklist questions

- Does the game involve non-players? How does it do it?
- Does the game have activities where players need to find out who the other players are?
- Does the game generate/use content that is based on other (non-player) people?
- Does the game use actors for non-player characters?

B.13
Conformance to physical and social settings

This feature relates to ethical and privacy concerns, conforming to social conventions, and adequacy to physical settings.

Related features: Mobility, Involving non-players

B.13.1
Motivations

Designers of pervasive games must be careful when design game activities that rely on personal information, as this raises ethical and privacy issues.

For example, many pervasive games rely on location information for game-play. Players might be scared if the game exposes their location along with per-sonal identification details. Sotamaa (2002) points out that the players might use gaming information for other ends, such as surveillance. In his example, a player whose friend has a *Botfighters* account might use this to monitor where he is.

B.13.2
Disturbing others and embarrassing situations

Another aspect of game activities, related to social concerns, is whether they disturb or interfere with non-players. For example, a game activity might play a sound on the phone, which might be disturbing due to the volume or content (like offensive words, etc.). This can be embarrassing to players, non-players, or both.

Some pervasive games use tangible game objects. Players might interact with those objects by using gestures. Depending on the gestures, players might get embarrassed with using the device on public places.

Another example comes from the game *Insectopia*, where players seek "insects" generated from Bluetooth devices. As Peitz and co-authors (2007) point out, as Bluetooth devices uniquely maps to a specific insect, players might know when someone is associated with a valuable insect, and then start stalking this person to capture it. The stalked person probably has no idea that he is the content source of a game, and certainly will become very scared.

In fact, game activities have to take into account the social setting where the game is going to happen.

B.13.3
Physical setting

Another issue is whether game activities are adequate for the places where they are going to happen.

For example, our *The Audio Flashlight* prototypes rely heavily on sound. On some public places, where it is noisy, playing the game becomes difficult. Other games suffer from this, as *Pirates!* (Björk *et al.* 2001).

Other possible examples include playing on outdoor areas, with direct sunlight. If game activities require players to look at the mobile phone screen frequently, under this situation the game experience might be downgraded.

B.13.4
Safety issues

As pervasive games often rely on mobility, designers should be careful in designing activities that do not expose players to dangerous situations, like physical harm.

In event games, like *Uncle Roy all around you*, keeping players safe is one of the duties of in-game management (see Section B.7).

B.13.5
Checklist questions

- How does the game handle player privacy?
- Do game activities possibly disturb non-players?
- Do game activities expose players (or non-players) to embarrassing situations?
- Do game activities expose players to possible dangerous situations?
- Do game activities conform to local social conventions/etiquette?
- Are game activities adequate to the physical setting of the game?

B.14
Usability

This feature corresponds to traditional usability issues from Human-computer Interaction, for mobile phones. Usability is a big research field per se, and detailing it is out of the scope of this work.

However, an aspect that caught our attention relates to designing activities that do not require players to constantly look at the mobile phone screen. This relates to using multiple modalities for interacting with the player. For example, besides visuals games may explore audio and haptics feedback.

This especially concerns pervasive games as in many activities, players might be moving. In this case, looking at the device screen might risk their safety, or may make them miss part of the game experience. Some authors refer to this approach as "design for 'heads-up' experience" (Ballagas and Walz 2007). Here are some examples of this concern in pervasive games:

> Feeding Yoshi – "… An audio alert is also made when a plantation or Yoshi is detected so that the player does not have to continually visual attend to the PDA screen". (Bell *et al.* 2006)

> Songs of North – "One of the main goals for the sound design of Songs of north is to provide information about the game situation so that the players can make some decisions in the game without looking at their device. Since the game involves moving, visual attention should be reserved primarily for safe orientation, and players should not be required to look at the mobile device when they are on the move." (Ekman *et al.* 2005)

REXplorer – "One criticism that we received early in the design process was the concern that people would have their attention focused primarily on the device screen, and that this would detract from the real attraction of Regensburg: the medieval architecture." (Ballagas and Walz 2007)

B.14.1
Checklist questions

- Do game activities require players to focus on the device screen too much?

- Do game activities use various modalities to interact with the player?

B.15
Persistency

Persistency refers to maintaining game state to be accessed through different game sessions over time. It directly influences several pervasive features, as Local space redefinition (*Hitchers),* Daily life interleaving *(Insectopia, Day of figurines*), Game autonomy *(Insectopia),* Social communication (games that form communities, as *Mogi, Gbanga Famiglia*). It also relates to Connectivity and Mobility, as players need to access game data from a variety of places, and possibly moving.

This feature is a requirement for games that aim at simulating a parallel world that evolve by itself. This includes all games that have social networking or community aspects.

Not all games require persistency. For example, event-games (*Can you see me now?, Manhattan story mashup, Uncle Roy all around you) and pervasive games that rely only on context-awareness (our prototype Pervasive Word Search).*

B.15.1
Checklist questions

- Do game sessions depend on previous sessions or stored data?
- Does the game support internal social networks or communities?

B.16
Connectivity

This feature influences all pervasive features where obtaining remote information may be required. This includes Mobility, Local space redefinition, Daily life interleaving, Social communication, Game autonomy. It is related to Uncertainty handling policy (as Connectivity is not reliable/available everywhere) and Game pacing (the pacing might influence Connectivity behavior).

Connectivity may exist in global or local scope. Global scope refers to connecting to remote peers that are not in the same physical place. Local scope refers to connecting to peers that are co-located.

Connectivity scope directly relates to the space scope of the game. Global connectivity makes it possible for activities to happen in very different places, possibly very distant. The game space scope is possibly big. Local connectivity restricts activities to happen in small places

B.16.1
Checklist questions

- What are the connectivity requirements for the game? Global? Local? None?
- How does the game handle uncertainties related to connectivity?
- What is the desired space scope for game activities?

Appendix C
Other prototypes

This appendix provides a brief description of prototypes developed during this research work: *Location-based Quiz Game*, and *The Audio Flashlight* series (5 games). The *Pervasive Word Search* prototype has been discussed throughout this research work. More detailed description is provided in selected publications or resources indicated in the text. This appendix also discusses how the pervasive game features (Section 5.1) are present (or not) in the prototypes.

All prototypes were implemented with Nokia smartphones, using the Symbian operating system.

The *Location-based Quiz Game* has been implemented in Java. The other games have been implemented with C++ and the Qt framework (Nokia 2011), along with extensions for Bluetooth (the QBluetooth library). The QBluetooth project (Valente and Ftylitakis 2011) corresponds to Bluetooth bindings for Qt that did not exist at the time of the projects were being developed. This had been a major issue that the author has contributed to mitigate by starting the QBluetooth project along with Nikolaos Ftylitakis, the original author of the library.

C.1
Location-based Quiz Game

The *Location-based Quiz Game* is a client-server game developed with Java ME (client) and Java SE (server). This game has been developed as a final project for the graduate course INF 2541 ("Introduction to Mobile Computing", Informatics Department) of Prof. Markus Endler, from the Post-graduate program in Informatics at PUC-Rio.

The goal of the project was to develop a location-based game for mobile phones, using the MoCA framework (Viterbo *et al.* 2008) developed at PUC-Rio.

The authors of the MoCA framework define it as *"a service-oriented middleware architecture that supports the development and deployment of distributed context-aware applications for mobile user"*. We have contributed to this frame-

work by implementing a version for Nokia Symbian phones of the Monitor component (LAC 2011) The Monitor is responsible for collecting context data from the mobile device and surrounding WiFi networks, and sending this information to a server running MoCA services. The *Location-based Quiz Game* client uses the MoCA Monitor, which has been implemented with native Symbian C++.

This project used one the MoCA services that provides a system for indoor localization through the WiFi network.

The game environment corresponds to a set of "rooms", which needs to be mapped prior to playing the game, with MoCA tools. In the *Location-based Quiz Game*, the game asks a question based on the room the player is located.

When the player enters a room, he receives an opportunity to answer a question. Each room has a set of questions (thematic), and the game draws questions from this set. If the player answers the question correctly, he is able to upload a question to the room. This question is valid only for the room where it has been submitted.

The game poses a limit on the number of questions a player is allowed to answer, when he is the room. When this limit is reached, the game asks the player to move to another room.

The questions are multiple choice questions. There are three possible alternatives, and only one of them is correct.

The players receive scores based on the total number of questions answered correctly, and also a ratio (correct answers x total questions answered). The player can also access a global ranking of players that informs this data. The ranking displays information of all players that are playing the game at the moment.

C.1.1
Pervasive game features

Table C.1 represents how pervasive game features are present in *Location-based Quiz Game.*

Feature	Description
Mobility	Restricted to the area that was previously mapped for the indoor positioning system.
Local space redefinition	Physical places become "rooms" in the game, which have special (and unique) content.
Game object tangibility	Not applicable. Mobile phones are mere terminals.
Game autonomy	Low. The physical game area needs to be mapped in advance for the game. It essentially an event game.
Game adaptability	Low. The game works only in the pre-defined area.
Involving non-players	Not applicable.
Game pacing	The pacing could be considered as "slow". Players have to walk in the (small) physical environment, and then stay there answering questions. There is no time limit for those activities.
Uncertainty handling policy (UHP)	As an older prototype, the game has no UHP. Uncertainties show up as game rooms not coinciding perfectly with the physical place. The positioning technology does not have notion of physical world limits. In this sense, sometimes it reported players as being inside a room, when they were outside the correspondent physical place (and vice-versa).
Daily life interleaving	Not applicable.
Device independence	Limited to Nokia smartphones. In theory, providing a monitor (and porting the client application to other platforms) is enough to make the game available for other mobile phone platforms.
Cross-mediality	Not applicable.
Social communication	Not applicable. It is a multi-player, but players do not interact with each other. They can know about other player by the ranking information.
Conformance to physical and social settings	Not evaluated in this game.
Usability	Not evaluated in this game.
Persistency	Requires persistency at the server side only. Players are able to post questions to a game room, so the game must store this information and present to players later, when they come back to the room. The server also keeps track of scoring for all players.
Connectivity	Client-server game that requires WiFi connections and a local wireless network.

Table C.1: Pervasive game features in *Location-based Quiz Game*

C.2
The Audio Flashlight series

The Audio Flashlight series corresponds to five games that were developed as research on non-visual games for mobile phones and context-awareness games. The last four games on the series are a variation of the first one.

The Audio Flashlight was he first non-visual game for mobile phones, developed as an exploration of non-visual mobile phone interfaces, using Semiotic Engineering principles (de Souza 2005) to design the user interface.

This section explores the games in this series.

C.2.1
General setting in the series

The player is inside a dark room and is not able to see anything. The player escapes the room when finding a treasure that is lying somewhere in the room. Figure C.1 illustrates a layout of a typical room of the game. The game does not display this map, though[102].
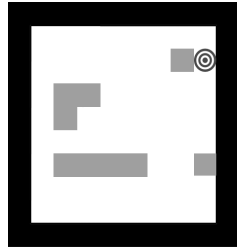

Figure C.1: A typical room in the game. Gray squares are internal obstacles and the bull's eye is the treasure location

The player has a tool to help him in locating the treasure: the audio flashlight. This tool works as an audio radar, playing music. When the player gets closer to the treasure, its music is faster and louder. When the player is far from the treasure, its music is slower and quieter.

The player has to manipulate the audio flashlight (device) to walk in the virtual world. To walk, the player has to tilt the device in four possible directions: forward, backward, left, and right. Figure C.2 illustrates the basic moves.
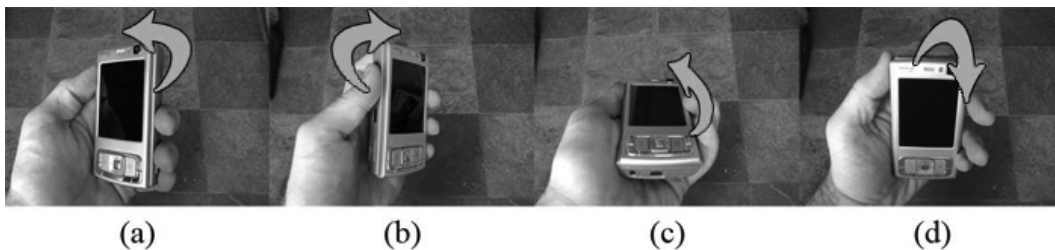

Figure C.2: Basic moves. Go left (a), right (b), forward (c), and backward (d)

The game acknowledges the movement by playing a footstep sound effect. The player can stay idle by positioning the device in the idle position (the device in the palm of the hand with the screen pointing to the ceiling), when the game

---

102 Except for version 5, where one of the players (the "Spoiler") is able to see the game map.

ceases playing the footstep sound. The player knows that he has entered a room when the game plays an audio effect of an opening door. The player may exit the game by turning the phone screen to the ground. Then, the game plays an audio effect of a closing door.

More detailed descriptions and research results for the first version of *The Audio Flashlight* are available in (Valente *et al.* 2008; Valente *et al.* 2009).

C.2.2
The Audio Flashlight 2

The second version is a two-player game, which is detailed in (Valente and Feijó 2011). The players are co-located, through Bluetooth, being the first multi-player non-visual game for mobile phones.

In this version, two players collaborate to find the treasure. The game map has a new element – the mines. If the players step on them, it is game over.

There are two roles in the game: the Flashlight and the Miner. The Flashlight plays like in the previous version. However, the Flashlight player is not able to detect mines. To solve this problem, the Miner comes in. The Miner is responsible for detecting the mines and alerting the other player about it. The Miner, however, is not able to move in the virtual world. In this sense the Miner is "piggy-backed" by the Flashlight. Both players play the game in close proximity, over Bluetooth. The Miner device acts as a "tactile window" to the game world. This window corresponds to a 3x3 grid and its contents depend on the location and movement direction of the Flashlight. Figure C.3 illustrates the possible configurations.
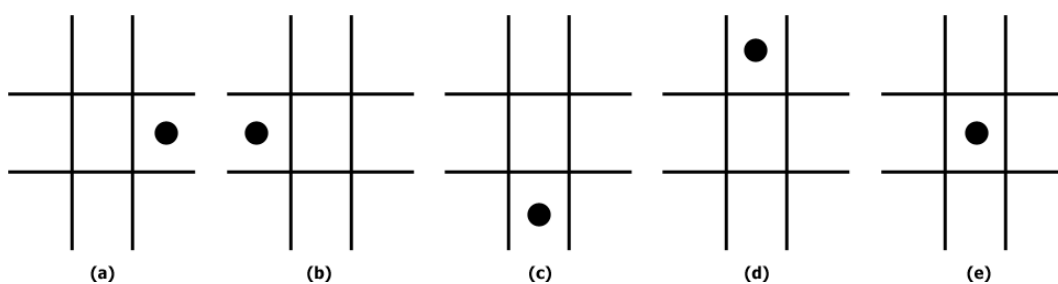

Figure C.3: Miner window configuration when the Flashlight is moving: left (a), right (b), forward (c), and backwards (d). The idle position is represented in (e).

The Miner is able to detect the mines by sliding his/her finger across the screen, and if there is a mine on a square, the screen vibrates at that square. The window contents change as the Flashlight player moves in the virtual world.

When the Flashlight is in the idle position, the Miner window allows the player to have a 360º perception around him/her. However, the range of this perception is limited to one square only. The Flashlight moves, the Miner window range of perception increases in the direction of movement. However, if the Flashlight moves like this, it can be more risky for the players, if the Miner is not able to detect a mine on time. The black dot in Figure 4 indicates where the players are relative to the Miner window. We added an aural sign to indicate where the players are relative to the window. When the Miner touches the square where they are, the game (in the Miner device) plays a sound effect. Notice, however, that the game does not display the grid or the black dot.

If the players step on a mine, a game over sound effect is played on both devices, and the games goes back to the intro.

C.2.3
The Audio Flashlight 4

The fourth version of *The Audio Flashlight* is a single-player that uses mobile phone sensors and alters the game map according to real world data.

The gameplay in this version is very similar to the first version of the game, but now the real world influences the game. We had some considerations in mind when designing this game:

- Have the player avoid playing the open areas, forcing him/her to go to "covered" places;
- Have a context-aware application;
- Make the player stay in areas where there are many people.

To make the player stay away from open areas, the application monitors the GPS signal. When the signal quality is "good" (under some criterion defined in the game), the game plays a noise sound effect that disturbs the player experience (*i.e.* the player is unable to hear the audio radar well). Moreover, the game dis-

plays a clock and a message like "take me to a safe place". If the clock runs out, the game is over. The clock always restarts when an "open area" is detected.

The game alters the game map according to the number of mobile phones around the player. In this version, the game map has some "doors" that the opens or closes depending on the results of the Bluetooth search. When the player collides against those doors, the game plays a sound effect to indicate that it is a door (different from regular walls and obstacles, which have haptics effects associated with them). The more Bluetooth phones found around the player, easier the game becomes, because the game opens more doors. Likewise, the less Bluetooth phones are found, more difficult the game becomes because there are more closed doors. The game indicates that it has become harder or easier by playing vocal sound effects saying that.

C.2.4
Some problems found

The prototype was implemented using the Qt framework, using a Nokia X6 and a Nokia N8 as test devices.

We noticed that the GPS sensor in those phones are unstable, meaning that there are a lot of noise in the data, so it is a bit difficult to infer that the device is in an open area just by looking at the location and precision data supplied by the sensor. There are a lot of factors that influence the GPS sensor data and precision, for example:

- Tall buildings around the device;
- Density and height of trees;
- If the weather is cloudy or sunny.

Currently, it is common that mobile phones use a variation of GPS known as A-GPS (Assisted GPS). In this form, the mobile phone uses the mobile phone network to get a first "guess" of where it is, so it is able to locate and connect to the satellites faster. This procedure greatly speeds up the initial connection with the GPS system, and also can be used if the signal reception is bad. However, this requires a working Internet connection. When the mobile phone uses the GPS

sensor in standalone mode, it might take up to 12 minutes to get a connection to the satellite, according to our measurements.

Using Bluetooth to change the game map is a nice thing, but it is necessary to consider some issues. For example, Bluetooth is not optimized for device discovery. In some devices, it may take some minutes for this process to finish, and not all devices in the neighborhood may be found on a single sweep. In this case, if the game needed the information about other devices in real-time, this solution would be unfeasible. Considering *The Audio Flashlight 4*, the fact that it is a nonvisual game helps in handling this Bluetooth limitation, as it is not necessary to display detailed and precise information about the devices in the vicinity.

C.2.5
The Audio Flashlight 5

The fifth installment of *The Audio Flashlight* is a multi-player game where three people play simultaneously, being an enhancement of version 3. The players are co-located – they play through Bluetooth.

The main difference from version 3 is that now the physical environment influences the game. As versions 3 and 5 are similar, they are described as one.

Different from the previous versions, now players compete among themselves, with distinct goals. There are three roles in the game: *Flashlight 1*, *Flashlight 2*, and the *Spoiler*.

With this version, we wanted to explore issues like:

- How to manage a local multi-player game with more than two players;
- Give players an environment that is slightly unpredictable, that unpredictability being part of the gameplay;
- How players react to unpredictability under pressure. In this case, the players have a time limit to complete the tasks;
- Introduce visual elements in the game (which did not exist in previous; versions of the game), while taking out others to make the game more challenging;
- Have elements from the physical world influence the game.

Players *Flashlight 1* and *Flashlight 2* play as in the first version of the game. They are still unable to see anything, being guided by audio and tactile feedback. Now they have a time limit to find the treasure. The one that finds the treasure first wins the game, and all three go to the next room.

The third player has the *Spoiler* role. This player is able to see the game map and the treasure location, being able to change its position. However, this player does not see the position of the other two players. The goal of this player is to disturb the other players' experience. If the other players do not find the treasure before the time limit runs out, this player is declared as the winner (and they start playing in the same room again).

The Spoiler is not allowed to change the treasure location all the time. There is another clock to tell when this player is allowed to do this. When this clock runs out, the Spoiler can change the location at his own will. However, once the treasure location is changed, the player needs to wait for the clock to run out again. Also, when there is a certain amount of time left on the main game clock (something like 30s), this player is not allowed to change the treasure location anymore.

In the current implementation, the main game clock lasts for three minutes, while the "treasure location change allowed" clock lasts for thirty seconds. When the main clock has only thirty seconds remaining, the third player is not allowed to change the treasure location anymore. This design decision was made so the third player does not have great advantage over the others.

The Flashlight players interact with the *wireless zone* – an area with a predefined number of WiFi access points. The current implementation defines a minimum of three access points to define the wireless zone. When inside the wireless zone, the Flashlight players earn a number of *freeze commands* – the exact number depends on the number of WiFi access points in the area. When a Flashlight player fires a freeze command is fired, the other two players are *frozen* – enter the *Freeze Mode*. This means the affected Flashlight player is unable to move, and his device starts malfunctioning. The Spoiler player becomes unable to move the treasure and to see the room map. The Freeze Mode lasts for a couple of seconds (10s in the current implementation), and after it expires the game comes back to normal for the affected players.

C.2.6
Pervasive game features

Table C.2 presents the pervasive game features in *The Audio Flashlight* series.

Feature	Description
Mobility	Not applicable for versions 1,2,3,5. For version 4, the game becomes easier if player goes to places with lots of Bluetooth devices. In this sense, it stimulates moving around. However, the game also works (although harder) if the player chooses not move.
Local space redefinition	None for versions 1,2,3. For version 4 and 5, this is subtle, although players probably may not perceive this.
Game object tangibility	In all games, the mobile phone is transformed in a tool called "The Audio Flashlight". The players manipulate the tool using gestures. For version 2, there is the phone transformed into a mine detector.
Game autonomy	Versions 1,2,3 do not use the environment, so it's not applicable. Version 4,5 can be played anywhere. The environment influences the game, but the game is playable without its influence.
Game adaptability	High. Only Bluetooth (v4) and WiFi (v5) are required from the environment. The accelerometer (versions 1,2,3,4,5) is not dependent on the environment.
Involving non-players	In version 4, non-players are passive from the game point of view. They participate in the game as sources of content, as their Bluetooth devices determine which door opens or closes. Not applicable in other versions.
Game pacing	Slow for version 1 and 4, and slower for version 2. Versions 3,5 are faster as there is a time limit to complete the main game goal.
Uncertainty handling policy (UHP)	• Hide (for all games) ▫ (versions 3,4,5) Bluetooth query is a slow operation, and search results may return false positives. Thus, the game does not require real-time response to Bluetooth data, and interacting with Bluetooth is performed indirectly. This applies to WiFi also. ▫ (versions 3,4,5) Do not represent information about Bluetooth or WiFi devices accurately. Instead, only announce important events ("game is easier", "entered zone", etc.). ▫ (all) Provide acknowledgments for all actions related to the accelerometer. If a gesture is recognized, provide an acknowledgment. If not, provide nothing. ▫ (all) Use well known device positions for the gesture commands (discretization): left, right, top, bottom, front facing.
Daily life interleaving	Not applicable.
Device independence	Limited to Nokia Symbian smartphones with Qt.
Cross-mediality	Not applicable.
Social communication	It is a multi-player game (versions 2,3,5). In v3,5, players may affect the experiences of each other, and they possibly play face to face due to the requirements of Bluetooth technology. For v2, players must collaborate to reach the goal. We expect the game to work as a means for heavy communication among the players.
Conformance to physical and social settings	Playing in public is hard due to noise. It is necessary to study this feature further in this game.
Usability	First evaluation for version 1 described in (Valente *et al.* 2008; Valente *et al.* 2009)
Persistency	Not applicable.
Connectivity	Uses Bluetooth networking in versions 3,5.

Table C.2: Pervasive game features in *The Audio Flashlight* series

Appendix D
Enhanced game design template

This appendix provides an example of a template for the game design document. We qualify this template as "enhanced" because it includes game design aspects (how the game is supposed to be, the rules, story, game flow, and other items) as well as elements derived from our methodology, which pertain aspects of Software Engineering. For example, this includes items for scenario specifications, using the domain knowledge and domain specific language this work proposes. This template has been inspired by the game design templates of the IPerG project (2008). The template is structured as follows:

Game name

Abstract

The abstract.

Last updated

Ex: July, 2011

1. Introduction

Introduction to the document, executive summary.

2. Game design overview

Overview to the game design, through the following Sub-sections.

2.1. Research goals

Goals for the research.

2.2. Project goals

Other specific goals for the software project, for example.

2.3. Game setting

General setting of the game, where it is played, etc.

2.4. Tools

Required software development tools, or related information.

2.5. Genre

Ex: "treasure hunt".

2.6. Target audience

Ex: "Causal players", "Age: 11, onwards".

2.7. Gameplay

A description of the gameplay.

2.8. Game session

Description of a typical game session.

2.9. Main features

Main features of the game, "selling points".

2.10. Other notices

Miscellaneous remarks.

2.11. Game modes

Ex: "single-player", "multi-player", etc.

2.12. Game goals

The main goals of players in the game.

2.13. Game environment

Where the game is played, for example, "virtual world, "physical/virtual worlds".

2.14. Story

Summary of the story, plot.

2.15. Visual theme

How the game looks, mock-ups of the user interface, etc.

3. <u>Game flow</u>

Summary of how players use the game. For example, how to start a play session, how a typical game situation is, etc.

4. <u>Content</u>

Mock-ups of the game content, artwork, if it is available.

5. <u>Detailing the game rules</u>

Breakdown of the formal rules of the game, if necessary.

6. <u>Technology</u>

Remarks about the technology the game uses, technological requirements, etc.

7. <u>Pervasive features</u>

How the pervasive features apply (or not) to the game.

Activity design

1. <u>Sensors and actuators</u>

List of sensors and actuators the game uses.

2. <u>Activities</u>

Specification of activities and use cases.