Mr. Michael J. Reese Jr., Johns Hopkins University

Michael Reese is the Associate Director at the Johns Hopkins Center for Educational Resources. Reese previously worked as an Educational Technologist at Caliber Learning and Booz-Allen and Hamilton. He also consulted with the University of Maryland School of Nursing to launch their distance education program. He earned an M.Ed. in educational technology from the University of Virginia and a B.S. in electrical engineering at Virginia Tech, where he was named the Paul E. Torgersen Leadership Scholar.

Dr. Peter H. Froehlich, Johns Hopkins University

Peter H. Froehlich is a Senior Lecturer in the Department of Computer Science at Johns Hopkins. Among many other things, he teaches compilers and interpreters, operating systems, systems programming, and video game design. Froehlich holds a diplom-informatiker degree from the Munich University of Applied Sciences in Germany, as well as a Ph.D. in information and computer science from the University of California, Irvine. His research focuses on the fuzzy intersection of software engineering, programming languages, and computer systems.

Ms. Joan Freedman, Johns Hopkins University
Introduction

This paper describes why and how an introductory video game design course was conceived and implemented, the best practices developed, and lessons learned since its first implementation in spring 2009. *Introduction to Video Game Design* was conceived at Johns Hopkins University as a creative, design-based course in which undergraduates work in collaborative, interdisciplinary teams on authentic projects. Students gained a broad knowledge of the gaming industry, along with practical development skills, through regular lecture-discussion sessions coordinated with weekly labs. The interdisciplinary student teams were mentored by industry professionals and worked throughout the semester to produce video games. The course development team faced a variety of challenges stemming from the inter-institutional, inter-departmental collaboration. The authors hope the case study described and evaluation data shared can help other schools planning to implement design-based courses, whether it is based on video game design or not.

The motivation for the course came from several emerging trends in higher education and the gaming industry. First, universities typically are slow to restructure academic departments to align with the new disciplines that define the jobs students will accept in the future (Frank and Gabler, 2006). Interdisciplinary courses provide a means for preparing students to work in industries that do not mirror the traditional departmental silos that incoming freshman usually find on college campuses. How often do engineering and computer science majors get to collaborate on projects with artists and writers where team members maintain their disciplinary expertise on the project? A convergence of understanding across what C.P. Snow calls the “two cultures” can be facilitated through authentic collaboration.

Second, educational research suggests students learn more when engaged through active learning techniques. As described by Michael Prince, active learning can take many forms – collaborative learning, problem-based learning, cooperative learning – and design courses provide an excellent opportunity for maximizing the use of these teaching methods. Unfortunately, many students do not engage in rigorous design courses until their junior or senior year (e.g., senior Capstone Design course). We wanted to provide students the opportunity to enroll in a design-course early in their undergraduate careers.

Third, and more specifically, while undergraduate enrollments in computer science have increased over the past several years, they have not recovered from the precipitous decline that began at the beginning of this decade. From 2001-2007, the number of U.S. students declaring computer science as a major decreased by 50%. The Johns Hopkins computer science department experienced similar declines, which led faculty to think about developing courses that would create excitement about the discipline.

Fourth, a large number of college students play video games. A recent survey of college students’ technology use reported that 43% of respondents play online, multi-player games. Numerous others play stand-alone games that can be downloaded from an app store onto their mobile device (Angry Birds had been downloaded over 250 million times by summer 2011). For several years, Johns Hopkins faculty received increasing student requests for a video game design course. Faculty also received inquiries from several video game design company executives working for companies in the Baltimore area about the possibility of offering a course
as a way to recruit talented students into this growing field. Projections suggest the gaming industry could grow into a $70 billion industry by 2015, providing employment opportunities for graduates during a depressed economy.8

Getting Started

In 2008, these factors led to the establishment of a curriculum design team with the goal of developing and implementing an introductory video game design course for undergraduates. The computer science department listed the course, but the team represented the interdisciplinary nature of the course created:

- Peter Fröhlich, lead instructor for course and senior lecturer in the Department of Computer Science.
- Michael Reese, associate director of the Center for Educational Resources (the Johns Hopkins faculty teaching and learning center on the undergraduate campus), acted as the project manager and instructional designer for the project.
- Joan Freedman, director of the Digital Media Center (Johns Hopkins student multimedia lab on the undergraduate campus), led the development of the lab section and course project.
- Tim Train, President of Big Huge Games (now Studio General Manager of Zynga East), helped coordinate input from gaming industry leaders during the needs assessment phase, provided feedback on curriculum design, and assisted with recruiting guest lecturers and team mentors.

The group defined the course objectives and designed a semester-long video game development project that would define the structure and topics of the course. The students would:

1. Develop a software system in small teams.
2. Design and implement a complete video game.
3. Plan, manage, and document a software development project.
4. Present status updates, project overviews, and technical topics.
These objectives facilitated the team’s development of an interdisciplinary, collaborative group project in which students created a working video game by the end of the semester. The course structure is described in more detail below starting with the course project which drove the design of the course. Appendix A contains a detailed listing of the topics taught in the lecture and lab section each week.

The Course Project: Overview

To assess students’ mastery of the course learning objectives, student teams were required to submit a working video game at the end of the semester. The curriculum development team debated whether to provide students with a detailed design specification for the game or to give them more design freedom by providing only open-design criteria. Some argued that open criteria would encourage students to be creative, whereas supporters of design specifications felt students new to game development would benefit from starting within a framework. An industry partner with teaching experience interviewed during the needs analysis phase suggested that students might be more creative when operating in a restricted design environment because they would have an existing scaffold on which to build. This was one of the most intensely debated issues in the course design phase. In the end, the curriculum development team decided to stipulate very few specifications believing that students coming from different disciplines would want more freedom to choose their development path and game genre. Those criteria were as follows. See Appendix B for the original project description.

Your game must:

- be fun
- be a game
- have a progression (e.g., levels, stages)
- include graphics
- include sound
- include a manual.

A short video describing the course includes an overview of three games developed in the course: [http://vimeo.com/5295315](http://vimeo.com/5295315).
The Course Project: Student Teams

To facilitate the interdisciplinary goals of the course, students were assigned to teams based on their skill sets and interest in gaming genres. Each of the 10 four-person teams included two “techies” – computer science or engineering majors to lead the programming – and two “creatives” – a writer, musician, or fine arts major. Johns Hopkins offers fine arts courses through its art workshops program but does not have a comprehensive visual arts curriculum; however, it is located near one of the top fine arts colleges in the country: the Maryland College Institute of Art (MICA). Students were recruited from MICA and the Johns Hopkins Peabody Conservatory, which trains students pursuing professional careers in music. The Peabody Conservatory is located on a separate campus, however, Peabody students may attend classes at both locations, which made recruiting these students feasible. Scheduling with MICA proved more challenging because of the different class schedules used at each institution. MICA courses are typically scheduled as 4-5 hour studio sessions. Courses at the Johns Hopkins undergraduate campus meet for 75 minutes twice a week or 50 minutes three times a week. The two institutions previously established an existing arrangement that facilitated the transfer of credit and tuition dollars.

Admission to the course was restricted by requiring faculty permission so the lead instructor could ensure a sufficient balance between writers, programmers, graphic designers, and composers were available for each team of four students. Students applied to the course by completing a brief survey asking why they wanted to take the course and what skills they could contribute to a team (See Appendix C). Demand exceeded course seats so the instructor included a four-person waitlist in case a student assigned to a team dropped out in the initial weeks of the semester. Students were then assigned to teams by the instructor.

The Course Project: Industry Mentors

Considering this was the first design course for many of the students, the curriculum development team felt it was important that students work with an experienced designer on the
The lead faculty assigned a staff member from a local gaming company to mentor each team. The mentor provided technical and design advice to students. The teams met with their mentor once a week to review their recent work, ask for advice, and define deadlines for the subsequent 1 to 3 weeks. Meeting weekly helped to keep students on task, but the student-industry relationships forged also produced unexpected benefits. One mentor shared an audio library with a team. Another recruited a colleague to provide advice on graphic design. The mentor structure proved to be one of the most beneficial aspects of the course and provided an important counterbalance to the open design criteria. Mentors helped students focus their video game designs and help teams set realistic goals for what they could achieve in a semester-long project.

One purpose of the mentoring relationships was to help students learn more about the gaming industry and help the gaming companies identify talented students for internships and job offers. The former was realized more than the latter. Four students were identified for internships as of 2011.

The Course Project: Monitoring Student Progress

While the industry mentor helped students set realistic milestones, project management responsibilities rested with the student teams. Each student team maintained a blog to document their progress and the challenges experienced, which the lead instructor could monitor. Industry mentors sent weekly email reports to the course instructor after meeting with their team to communicate additional feedback on each team’s progress. The mentors used the following five questions as a guide to their update.

1. Did everybody show up? Were they on time?
2. Did everybody seem involved?
3. Did they seem prepared with prototype, artwork, working code sounds, music, etc?
4. Did they make appropriate progress since last week? What are their major stumbling blocks?
5. Do they have problems with a specific area/technology that we need to help them with?

Four times during the semester, each student team gave a presentation to the class on their project. Within the first two weeks, students proposed their general game concept to the class for feedback. Subsequently Alpha, Beta, and Gold course presentations were scheduled every 3-4 weeks in which the teams received feedback on the current development of their game from their fellow students, instructors, and mentors. All final projects were presented in a final session held at the Digital Media Center (DMC), the campus student multimedia lab, in which the class and invited guests had the opportunity to play the games.

The Course Project: Student Peer Evaluation

Students completed a peer evaluation form for each of their teammates at the end of the semester. The faculty considered these when converting the team grade to an individual grade—assigning a higher or lower grade as needed. Students wrote a self-assessment to communicate their self-perceived contributions to the final product to ensure their perspectives were captured...
in a 360-degree evaluation. Students completed individual assignments throughout the semester including short writing assignments such as a review of a professionally-produced game.

The Course Design: Overview

The curriculum development team created the course structure and content after designing the course project so that class meetings would support students developing their video game. The team recognized the importance of including a lab section, along with a lecture component, to help students learn key multimedia development skills and become familiar with the lab environment where they would likely complete most of their development outside of class time. In addition, the lab provided an opportunity to explore general creative design topics. For example, students at the first lab session defined rules for using a bag of mixed materials as a game (e.g., cards, dice, timer, clay, action figures). Appendix A contains a detailed listing of the topics taught in the lecture and lab section each week.

The Course Design: Lecture

Students met in lecture three days a week for 50 minutes. The purpose of the lecture component was to expose students to introductory gaming topics (e.g., gaming genres, gaming history, intellectual property issues, computer graphics principles, introductory programming, business of gaming). The goal was to help students gain a broad understanding of how the industry worked, while also familiarizing students on fundamental skills their teammates would contribute to the project. The purpose was not to train students to mastery, but to help students better understand the roles and skills of their respective teammates. For example, musicians would need a basic understanding of how programmers write code that manipulates music files while programmers would need a basic understanding of how audio files are created and published.

Because of the broad number of topics presented in the lecture, the instructor of record and primary lecturer, Dr. Peter Fröhlich, invited guests from industry and other universities to lecture. Ken Rolston, a highly-regarded game designer who has authored both video and paper-based games discussed the importance of role-play in designing a game. Katie Hirsch, a designer from Firaxis Games and an instructor at the University of Maryland Baltimore County, discussed user interface design. Tim Train talked about how to prepare for a career in the gaming industry.

The Course Design: Lab

The associated lab section in which student learned the basic skills for designing a game was hosted by the Digital Media Center (DMC). Students met weekly for 75 minutes. Topics included work flow planning, character development, 3D modeling, animation, titles and transitions, sound effects. Again the purpose was not to teach to mastery but to introduce teams to project management concepts, design techniques, and development platforms. Student teams decided which technologies they would use to build their game and gained a deeper mastery on their own. DMC professional and student staff were available to help students outside of class.
The Course Design: Additional Class Resources

Play was considered an important component of the learning experience. The team reached out to the Johns Hopkins Sheridan Libraries to establish a modest gaming collection. The professor then incorporated play-and-review assignments as course homework. While many students owned games, the instructor felt it was important for them to have the opportunity to play games from different genres and systems. The librarian for computer science joined the development team and led the game collection development effort. She also helped the team navigate the difficult process of establishing new library borrowing privileges and policies for game media because the existing policies for traditional audio-visual materials (e.g., DVDs, VHS tapes, CDs) did not match the users’ needs for borrowing video games. The library did not purchase consoles for students to play the games as these were available in the DMC.

In addition to teaching the lab section of the course, DMC staff developed two connected spaces to support game design, prototyping, testing, and research: the Gaming Lab and Gaming Lounge. The JHU Gaming Lab is a fully equipped development and testing lab created to support course-related and independent game development and exploration by students. The lab contains high-end workstations, including NVIDIA graphics cards that were acquired through a generous donation from the company and an array of cutting-edge development software, as well as a large-format Samsung screen and 5.1 surround-sound system for testing and game play.

![Figure 2: Gaming Lab (Photo Credit: HPS, Will Kirk)](image)

The Gaming Lounge is a student-friendly space that features comfortable furniture attractively arranged around a series of gaming stations. Game stations include: a flat-panel display with parabolic speakers for directed sound that can accommodate a number of different game consoles; a classic arcade-style gaming table; and a beta-test kiosk that can be used for console
play or to display student-developed games. In response to increased interest in gaming among students, the DMC also sponsors a number of game-related events. Networked gaming nights are held four times per year, during which all 18 of the DMC’s workstations are converted to a cyber-café for networked multiplayer gaming.

Figure 3: Gaming Lounge (Photo Credit: HPS, Will Kirk)

Lessons Learned: Evaluation Data

The curriculum development team conducted rigorous formative assessment as part of a thorough evaluation program during the first implementation to improve the course during its maiden voyage and for future semesters. These activities included collecting student feedback through interviews, focus groups, and anonymous online surveys. The most relevant data to a general audience is summarized below.

Overall, students liked the concept of the course. The video game design project was communicated as their favorite aspect of the course because it allowed them to work in teams and to create a product. One student commented, “Making a game was cool!” Students said they learned to appreciate the diversity of talents of their teammates. During a focus group, a computer science major said, “Artists contributed a lot. They are amazing!” while an artist commented, “What computer scientists do is humbling.” In fact, several students felt they learned more game development skills from each other than from lecture and lab. Figures 4-6 summarize students’ responses to an online survey about their experience working on the course project.
Figures 4-6: Student Feedback on Course Projects (n=15)
Students appreciated the exposure to industry experts as mentors and guest speakers (see figure 7). Several teams remarked how valuable their mentor’s advice had been. An unplanned but important learning experience arose in the first implementation of the course when one of the companies that supplied mentors underwent a major downsizing in which the company laid off 1/3 of its staff. This unfortunate event showed students firsthand the effects of the Great Recession. Many talked directly with their mentors about the experience. While the experience did not significantly deter students from wanting to work in the gaming industry (see figure 8), they recognized the volatility of working in the entertainment sector.

As described above, the curriculum development team debated using detailed design specifications vs. open-design criteria. While the latter was chosen, students preferred the former. They articulated their preference in several ways. First, students communicated that the general topics in the lecture course did not provide enough detail, nor were they always relevant or provided in time. For example, audio editing was taught after the first alpha release in which
some teams wanted to integrate sound. By using detailed design specifications, the lecture topics could be better aligned with development deadlines. Second, students felt an introductory course should use narrow design parameters that would focus their attention on the basic skills of video game development. Students felt creative opportunities for designing their own game could be facilitated through advanced game design courses (e.g., a Capstone Design course). In subsequent implementations, the course project requirements were narrowed to restrict the development framework to Python/PyGame and students were required to create a specific type of game (e.g., top-down space shooter, side-scrolling platform like Mario Brothers). Finally, students wanted more frequent project deadlines than the three Alpha, Beta, and Gold presentations. Students felt a synchronized development schedule with weekly assignments would provide more structure to their development timetable and facilitate more exchange between teams. Students now complete 10 course assignments over the course of the 14 weeks that guide their game development. Related to this suggestion, students wanted more opportunities to demo their projects during class. They appreciated feedback from peers and wanted more to assist them in refining their game design and play. Students suggested they review each others’ games every 2-3 weeks and link these demos to submission deadlines.

The students also recommended reorganizing the course structure so as to emphasize labs over lectures. Students felt the labs were too short and they did not get enough time to apply concepts under the instruction of the lab teacher. Student teams comprised of members from different campuses and schools also wanted more open time in labs to collaborate because they experienced difficulty scheduling common times to meet. DMC staff also felt that a 75 minute lab session was not enough time to conduct adequate hands-on training. Based on this feedback, the professor tried using online lab modules in the second implementation of the course, but students still felt they were not specific enough for their game development requirements. In the most recent implementation, the course lecture has been reduced to two days per week and students are required to meet at the DMC for a 3-hour designated meeting time during which they work on their game under the supervision of the course teaching assistant.

Another suggestion was to offer two tracks, one on game programming and the other on artistic development for gaming. Students would enroll in the course track most aligned with their expertise (programming vs. creative arts) and then meet during labs to work together on projects. Students also suggested establishing open, self-paced labs in which they would receive credits for demonstrating competencies they master. This would enable them to learn the specific skills needed for their project. This model has been used in a game design course at Ball State, but was not adopted at Johns Hopkins because coordinating common projects across two courses would be too difficult.9

Though students liked the interdisciplinary teams, some groups reported conflicts and difficulty communicating ideas across disciplinary boundaries. They requested more instruction on managing team dynamics and project management. Faculty from other departments who teach courses with team projects offered to share materials and assignments used to prepare students for working in teams, including team contracts established at the beginning of the semester.
Summary of Best Practices

We formulated the following best practices that can be generalized to any design course based on our experience implementing this course and responding to the challenges that arose.

1) Provide students detailed design specifications. For a freshman or sophomore design course, detailed design specifications help students develop a workable project without limiting their creativity.

2) Communicate frequently. Weekly feedback from a mentor/teaching assistant was highly appreciated by students and ensured they made continual progress during the semester.

3) Implement team contracts. Establishing good working relationships and workflows at the beginning of a collaborative student project pays big dividends. Team contracts written and signed by team members in the first weeks facilitate good working relationships throughout the semester. Contracts have not been used in this course, but are being considered for the future.

4) Establish industry connections. Assigning relevant industry mentors provides students with opportunities to gain insight into authentic practices. Mentors find that they gain from these relationships as well. Internships and summer job can be established or additional staff expertise can be leveraged. Mentors do not have to meet weekly to be productive, but regular meetings throughout the semester provide more opportunities for students to learn from those with relevant expertise and experience.

5) Include adequate lab experiences. The lab section exposed students to relevant development technologies and provided students hands-on experience developing different components of a game.

The authors would like to thank the Smart Family Foundation, NVIDIA Corporation, and Valve Incorporated for the resources they donated to support this project.

References


Appendix A – Course Schedule (Weekly Topics for Lecture and Lab)

Week 1
Lectures: Welcome, Brief History, Being Successful in the Industry
Lab: Design and Planning, Time Management, Storyboarding and Flowcharting (some combination or reduction of these topics)

Week 2
Lectures: Archetypal Video Games (Genres)
Lab: Understanding and making rules (design rules for ‘board’ game)

Week 3
Lectures: Intro to Programming
Lab: Image Editing / Character Design exercise (2D photoshop/illustrator/flash)

Week 4
Lectures: Intro to Graphics
Lab: Sound Effects/Basic Foley

Week 5
Lectures: Modular and Object-Oriented Programming
Lab: Animation (principles and basic technique)

Week 6
Lectures: Three Dimensional Graphics
Lab (proposed): 3D Studio (3D Studio Max, blender?) – modeling

Week 7
Lectures: User Interface and Interaction Design
Lab (proposed): 3D Studio (3D Studio Max, blender?) – textures

Week 8 -- Spring Break

Week 9
Lectures: Environment/World/Level Design
Lab: Character Design / 3D (Poser)

Week 10
Lectures: User Interface and Interaction Design
Lab: Controllers (Wiimote demos, other unusual controller configuration?),

Week 11
Lectures: Physical Models and Game Physics
Lab: Game Soundtracks
Week 12
Lectures: Serious (Learning?) Games
Lab: transitions, title screens etc

Week 13
Lectures: Social Impact of Games (violence, ethics, social change)
Lab: art games

Week 14
Lectures: Social Impact of Games (literary form, propaganda, gender and race)
Lab: open

Week 15  Party - demo games
Appendix B - Description of Assignment

One major component of 600.255/600.256 is the semester-long team project in which you will produce an experimental video game. Note that "experimental" does not mean "incomplete," but rather that we do not require you to produce a game at the level of sophistication you see in the industry today. So while we would be thrilled if you produced a game like Urban Terror or Call of Duty 4, we'll be quite happy if your version is 2D instead of 3D, features 3 instead of 10 types of weapons, has 2 instead of 20 hours of game play, supports play on only one machine instead of across a network, etc. Here is what we're looking for in more detail:

Your game must be fun. This is the most important ingredient, and it's the most difficult to describe and achieve. A good way to ensure fun is to follow an established pattern that others have shown to be fun and improve on it. If you don't want to follow a pattern, you will have to do lots of play testing while you develop the game. Other teams will give you feedback on your game at certain points during the course, but you probably want to run your own play testing sessions as well to make sure you're building a fun game.

Your game must be a **game** and not some other kind of interactive experiment. It **must have objectives** for the player to accomplish, it **must have obstacles** the player has to overcome, it must support some kind of **progression in difficulty** that can realistically be mastered in several hours of play, it **must have loading/title screens and menu-based configuration**, it should provide some way for players to **review statistics** of how well they and others did.

Part of the "progression" of your game should be a **variety of levels or stages**. Think different mazes for a Pacman-type game, different flying patterns for a Galaxians-type game, the various levels of Doom or Quake, and so on and so forth. Challenge stages, boss levels, everything that adds some variety to the game (variety that the player will look forward to) is good.

Your game must be **graphical in nature**, text adventures and other text-based games are too prehistoric at this point. Whether your graphics are 2D or 3D is secondary. Whether your models/textures/images are fancy or plain is secondary. Of course 3D and fancy is cooler, but you can get an A with 2D and plain as well. If your game requires state-of-the-art graphics hardware, it would be good to make level of detail and texturing configurable to support a wider range of machines.

Your game must have **sound effects and a musical score** of some kind. Ideally the score is built around a recurring theme but has variations according to the mood you want to create for a given level or stage in the game. A more involved title/ loading score combined with various background scores throughout the game and maybe a sad ending score when the player fails are all good ideas. Make sure that the volume of score and sound effects can be adjusted by the player.

Your game must come with supporting materials. A **manual is a must**, but depending on the kind of game you produce it can be brief. It should cover the basic **plot, installation procedures, and basic configuration** for sure. It would be great if you could design a **retail box** as well.
However, don't spend too much of your time on marketing materials, the game itself is more important in the end.

We will do our best to give you feedback on your games regularly. You'll have a mentor with whom your team will meet once a week to discuss how things have been going. You'll have the opportunity to review the games other teams are working on. You'll have the opportunity to review the games other teams are working on. Your game will get reviews from other teams as well. We'll also have at least one public play test session in the DMC that will be advertised widely to get you some outside feedback. In the end, it's all about a fun game with decent graphics and sound and you should be good to go.
Appendix C - Intro to Gaming Course Skills & Interest Survey

The information you provide on this skills and interest survey will help Prof. Fröhlich assign students into teams for the course project. Please complete this survey by Friday, January 23rd.

1. Name
2. Email
3. Major/Minor
4. School/Year (e.g., Engineering, Jr., MICA 4th years)

5. In 300 words or less, please describe your experience (if any)
   * designing games;
   * programming;
   * writing creatively;
   * writing or producing music;
   * developing multimedia resources; and/or
   * creating art.

6. Please list the title of any courses or workshops you took in the following areas along with where you took them.
   * programming
   * arts (drawing, sculpting, etc.)
   * writing
   * music (composition, instruments, electronic, etc.)
   * multimedia development (animation, film, etc.)

7. Choose the top 3 game genres you like to play.
   action
   adventure
   construction/management
   role-playing
   strategy
   vehicle simulation
   puzzle
   sports
   traditional board games
   other (describe)
   Other, please specify

8. For the course project, choose your top 3 choices for the type of game you want to make.
   action
   adventure
   construction/management
   role-playing
   strategy
   vehicle simulation
9. Please rank your programming skill in the following languages.
   1-No Experience  2  3  4  5-Highly Skilled
   C++
   Java
   Perl/Python/Lua/JavaScript
   HTML

10. Please describe your experience with assembly or other languages if any.

11. Please rank your familiarity with the following operating systems.
    1-No Experience  2  3  4  5-Highly Skilled
    Windows
    Unix
    Mac

12. Please rank your familiarity with the following APIs (application programming interfaces).
    1-No Experience  2  3  4  5-Highly Skilled
    Java AWT/Swing
    OpenGL
    DirectX
    SDL/PyGame

13. Please rank your familiarity with recording and editing audio/music using the following applications.
    1-No Experience  2  3  4  5-Highly Skilled
    Audacity
    Pro Tools
    Garage Band

14. Have you used other applications to record or edit audio? Have you ever played or recorded with a band? Please describe below (including what instruments you play).

15. Please rank your familiarity with recording and editing video using the following applications.
    1-No Experience  2  3  4  5-Highly Skilled
    Adobe Premiere
    iMovie
    Final Cut
16. Have you used other applications to record or edit video? Please describe below.

17. Please rank your familiarity with the following multimedia applications.

1-No Experience   2   3   4   5-Highly Skilled

Highly Skilled
Adobe Dreamweaver
MS Front Page
Adobe Illustrator
Adobe PhotoShop
Adobe Flash
Adobe After Effects
Poser
3D Studio max
Maya
Google SketchUp

18. Please describe your familiarity using any application not listed above (be sure to list the name of the application and what it does).

19. Please rank your experience doing the following things.

1-No Experience   2   3   4   5-Highly Skilled

Drawing
Painting
Modeling and Sculpting
Making Stuff (e.g., crafts, welding, hacking electronics, customizing objects)
Creative Writing

20. All course project teams will be assigned a mentor who will meet with the team weekly to check on their progress. Mentors will include JHU faculty and staff along with gaming industry experts. Mentors from the gaming industry are interested in working with student teams to identify potential interns, however, this may require the team to occasionally travel to Hunt Valley (just north of the city). Please indicate if you would prefer a mentor who is from industry or from JHU.

JHU Mentor
Gaming Industry Mentor
Don't Care

21

Any other information you want to share?