SOFTWARE QUALITY ASSURANCE

PRESENTED BY FREDERIKA EDGINGTON-GIORDANO

11/9/2019

THIS IS NOT HOW YOU QA GAMES…

- https://www.youtube.com/watch?v=BRWvfMLI4ho

DISCLAIMER

- This talk is solely my own and any opinions expressed here are not a reflection of any of my current or previous employers.

SR. QUALITY ENGINEER AND GAME DEVELOPER

- TVGS Email: education@techvalleygamespace.com

MY EXPERIENCE

- Quality Assurance/Engineering
  - Worked with developers on-site and remotely.
  - UAT, Regression, SDLC.
  - Tested new and old software.
  - Tested on mobile devices and on PC.
  - Web apps, mobile apps, SAAS.
- Game Development
  - Some Unity3D
  - Construct 2 and 3
  - Ren'Py

QUESTION FOR AUDIENCE

What is Quality Assurance?

WHAT QA DOES

- QA/QE does **not** break anything!!!
- We find where the developers broke the data or the code or the game mechanics…
- We are a safety net for everyone's mistakes, miscommunications, and misunderstandings – the key to our work is communication!
- Ability to pay attention to details, but also keep the larger picture in mind.
- We constantly need to learn and pay attention to the systems we are testing.

STAGES OF QUALITY

- Requirements/Initial Design/Game Idea
- Coding/Creating game
- Testing for bugs
- User feedback – “Playtesting”

APPROACHES

- Large scale to small scale
- Look for patterns
- Look for boundaries, extreme examples
- Isolate the causes/triggering events
- **Always** fail the game/software as quickly as possible!

THE MOST IMPORTANT PART: COMMUNICATION!!!

- Clear understanding of requirements.
- Clearly defined tests cases/scenarios.
- Create accurate and helpful Bug reports.
- Ask questions!
- Keep lines of communication open during the entire process.
- Know what people in different roles are talking about.

TEST CASES/SCENARIOS

- Clear reproducible steps
- Any needed inputs, files, urls, character types, etc.
- Actual results
- Expected results
- Iterate test cases by updating as needed
- Cover all testable requirements
- Include Edge cases

QUALITY ASSURANCE AND GAMES (ALSO SOFTWARE)

- Identify and test the places where things will go wrong.
- Get familiar with bugs and you may see them again.
- Come up with more tests as you go.
- Do some crazy stuff, because users will do crazy stuff.
- Do other crazy stuff, because it will bring out a problem more efficiently.
- Overall, interact with the game systematically and find bugs.

BUGS

- Description of what occurred (Why/how is this wrong or broken?)
- Steps to recreate the bug
- Evidence of what happened ('It worked on my machine…')
- Relevant context (url, environment, settings, etc.)
- Impact
- Bug Tracking – Excel, Xray for Jira, Trello cards

PITFALLS THAT TESTERS CAN FALL INTO

- Getting used to the broken system.
- Test environments and data can be unrealistic (this can go both ways).
- Things that were fixed the last build are now broken (again.)
- It can be hard to break the bad news, especially when a project is past the deadline.

BENEFITS OF QUALITY ASSURANCE

- Better initial design (and fewer changes later in the process)
- Game handles user behavior other than the ‘Golden Path’
- Updates and changes are easier
- User feedback helps inform updates
- Save time and effort

Q&A

LEARN MORE

Extra Credits

- QA – https://www.youtube.com/watch?v=ntpZt8eAvy0
- Buggy Games - https://www.youtube.com/watch?v=s1_50T5GwZ8