

# UX in Games

Why UX is useful  
& some tools for UX design

Who is this guy?

# Ben Humphreys

- Studied Computer Science at university in the UK
- Worked in video games industry for ~6 years
  - Gameplay programmer
  - UI programmer/designer
- Self-taught UX designer
  - Interested in web design, typography
- Worked as UI programmer on **Vampyr** at Dontnod
  - Action-RPG for PC & consoles
  - Lots of RPG menus, supported controller & mouse/keyboard



# CHARACTER

Jonathan Reid

# LEVEL 15



Power & Mesmerise Level



## MAIN HAND



## OFF HAND



## SERUMS



## STAMINA



## LIFE



## BLOOD



## PASSIVES



## ACTIVES



## ULTIMATE



# Ben Humphreys

- Currently working on **Industries of Titan** at **Brace Yourself Games**
  - Cyberpunk city-building game
  - Coming out Q3 2019!
- Brace Yourself Games?
  - Crypt of the Necrodancer
  - Industries of Titan
  - Phantom Brigade
  - More games soon!

- (come work with us!)



# Ben Humphreys

- Currently working on **Industries of Titan** at **Brace Yourself Games**
  - UX designer/programmer, game designer
  - Industrial city-building sim/strategy









# SHIP DESIGNER

Clear All Devices Delete Mode

## DEVICES

## ENERGY

- GENERATOR 1,000
- SMALL BATTERY 500
- MEDIUM BATTERY 1,500

## WORKERS

- COCKPIT 200
- BED 200
- LIFE SUPPORT 500
- MEDICAL POD 1,000

## OFFENSE

- LASER CANNON V2 1,000
- LASER CANNON V4 2,000

## DEFENSE

- HARD SHIELD 2,000
- AUTOMATED SHIELD 2,000



**BED**  
 200/99  
 Crewmembers start here in combat. Each crewmember has a bed. Placing beds increases the number of crew on your ship.  
 Move Delete Del

**CHECKLIST**  
 Required items for takeoff.

- FUEL
- ENGINE
- POWER
- LIFE SUPPORT
- COCKPIT
- 2 WEAPONS
- 3 BEDS

**LAUNCH**

Oxygen

El	000
Kaff	000
Las	000
Las	000
Las	000
Las	000

140/140 32

2,200/2,200

Weapon Slots: Laser Cannon v2 1, Laser Cannon v4 2, EMPTY

Back to hull selection



Burst Laser 1

Charging

Precision Laser 2

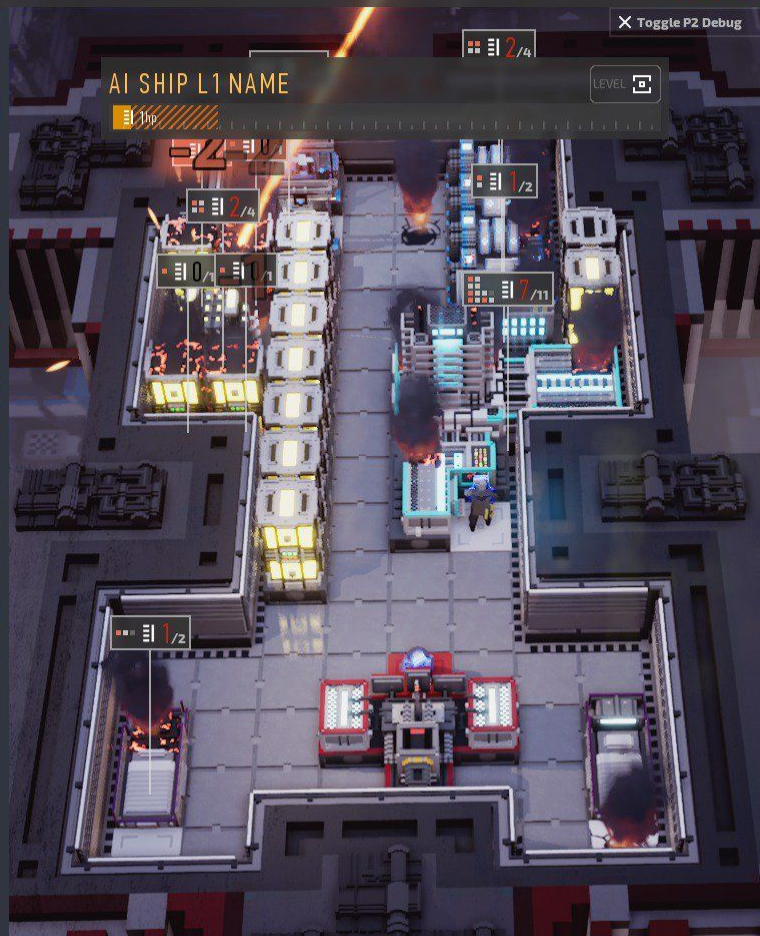
Charging

Precision Laser 3

Charging

Burst Laser 4

No Worker



Flee!

# Design is a doing word

- You can't get better unless you practice
- Today's lecture has audience participation!
- Get out pen & paper, or pen & tablet, or something



*Design Time!*

Today's goal:

Get tools to think  
about UX design

Today's topics:

- UX > on-screen stuff
- Tools for UX design
- Check out some games

- UX > on-screen stuff
- Tools for UX design
- Check out some games

 x05



 21

000011300

 062



What state is being shown here?



Character

Lives

000011300

062



Score

Time remaining

21

Special coins

Coins



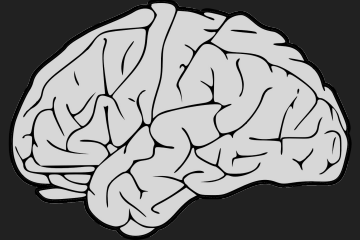


# What's the goal of UI?

`int numCakes;`



*42 cakes*

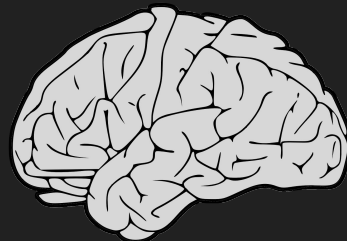


Put **information**  
into the player's **brain...**

```
int health;  
int maxHealth;
```



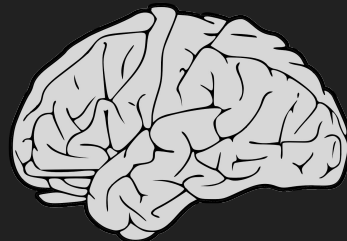
**142/176**  
*HP*



```
int health;  
int maxHealth;
```



**7/176**  
*HP*

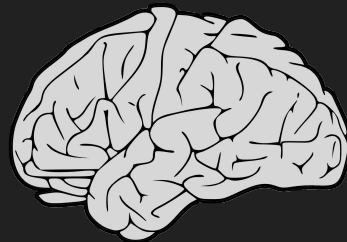


Is my health OK?

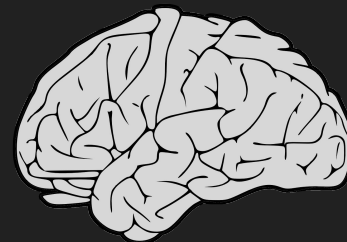
```
int health;  
int maxHealth;
```



7 HP



```
int health;  
int maxHealth;
```



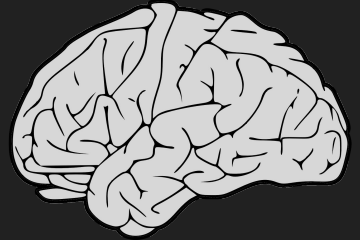
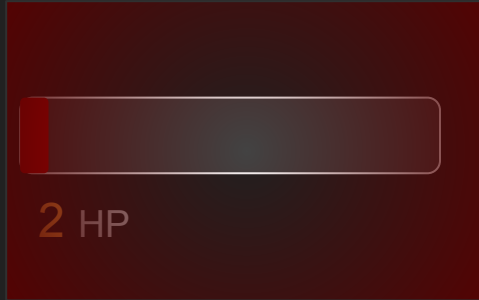
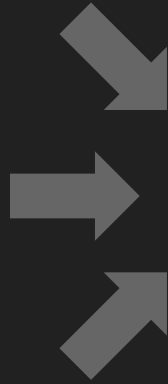
## Health-related **mechanics**

- Poison
- Enemy damage values
- Armor

## Desired **player experience**

- Don't worry about health until near-death

```
int health;  
int maxHealth;
```



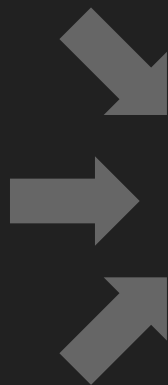
## Health-related **mechanics**

- Poison
- Enemy damage values
- Armor

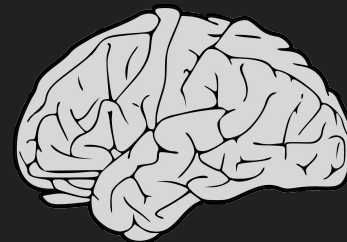
Desired Player  
Experience

Game State

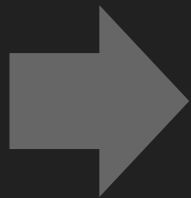
Game  
Mechanics



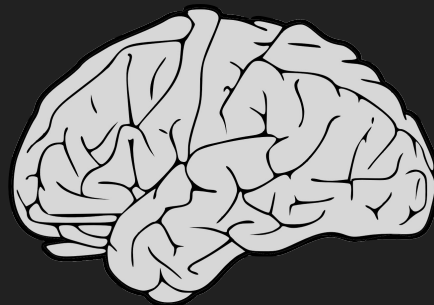
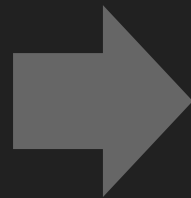
UI Visuals



```
int health;  
int maxHealth;
```



159 HP





# But wait, there's more

Gems are rare!

```
int numGems;
```



Gems are used for trading

Player can only  
carry 300 gems

Unless they have the  
Big Gem Bag...

Players can get Gem-itis,  
that drains gems

Gems can be broken  
into gem shards

# Game

*Get Coins!*

```
int numCoins;
```

Other Game Mechanics  
Max coins?

# UI

Real-world knowledge:  
“Coins can be exchanged for  
goods and services!”

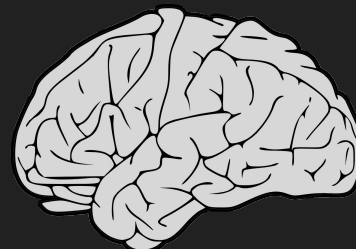


42

Gold is worth a lot  
Gold colour = monetary value

# Player

(not to scale)



*“I want to get  
these useful  
coins”*



Max 99 lives?



Max 3 special coins



Coins don't seem to have an upper bound?

000011300

062

Seems that score can go up to 999,999,999?

Time can max out at 999?



# Game Experience Considerations

- How much do we want players to worry about their health?
- Is health management core to the experience?
- If increasing max health is possible, how often does this happen?

# Designing without intention

- Internet image search for “health bar” ...





# Segmented Health

Used in action RPGs, platformers

- Want to give the player a **simple relaxed experience**
  - Avoid numbers where possible
- **Easy to understand** at a glance
- Easy to show small increases in maximum health
- Works well with object pickups
  - Pick up heart = increase health by 1



# Lots 'o Numbers

Used in RPGs

- Numbers get larger and larger as the player levels up
- Feed the **feeling of growing more powerful**

NAME	BARRIER	HP	MP	LIMIT	WAIT
Cloud		5320/5320	680		
Tifa		4543/4543	570		
Cid		4242/4242	535		



# Blood Mist Borders

Used in some first-person shooters

- Deliberately vague health indicator
  - Perfectly healthy is “no blood mist”
  - 0% health point is not clear
- Goal is to tell the player “get into cover”
- Works with regenerating health



# Partial-info indicators

Used in horror, classic survival games

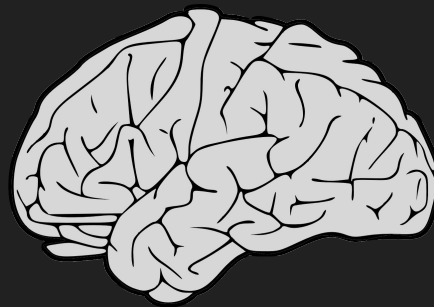
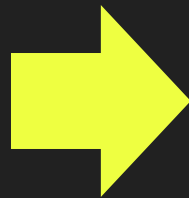
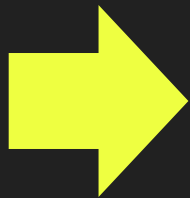
- Health is between 0~100, but game only shows partial information
- Want player to feel scared, unsure
- Healing items are rare, want the player to feel they have to get the most out of them
  - They will get into Caution/Danger, feels more risky & scary
- If players had perfect accuracy about healing item



# What we can do with good UX design

- Showing game state is only part of it
- We can do so much more
- Convey many **mechanics** through minimal UI
  - Player should **intuitively understand** what's **possible**, what could happen
- Control the players **experience**
  - Make them **feel** a certain way
  - Change their behaviour in-game

Game



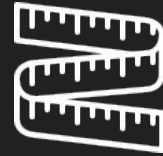
# Health Bar Design Exercise — Round 2

- Design a health bar
- Specification:
  - Third-person action-romp
  - You play as a giant octopus, rampaging through a city
  - You take damage from tanks armed with cutlery
  - You start with 8 legs
  - Can gain more legs, up to 16
  - You can take damage, losing legs, die when you hit 0 legs
  - You can gain up to 8 armour by grabbing buildings and sticking them to you

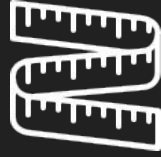
- UX > on-screen stuff
- **Tools for UX design**
- Check out some games

# Tools for UX design

- Unfortunately there's **no** “one weird trick”
- Instead, lots of **tools**
  - Not tools like software
  - Tools for your **brain!**
- Learn more of them as you get experience
- Make up your own!
- Learn when to **use** them
- Learn when to **ignore** them

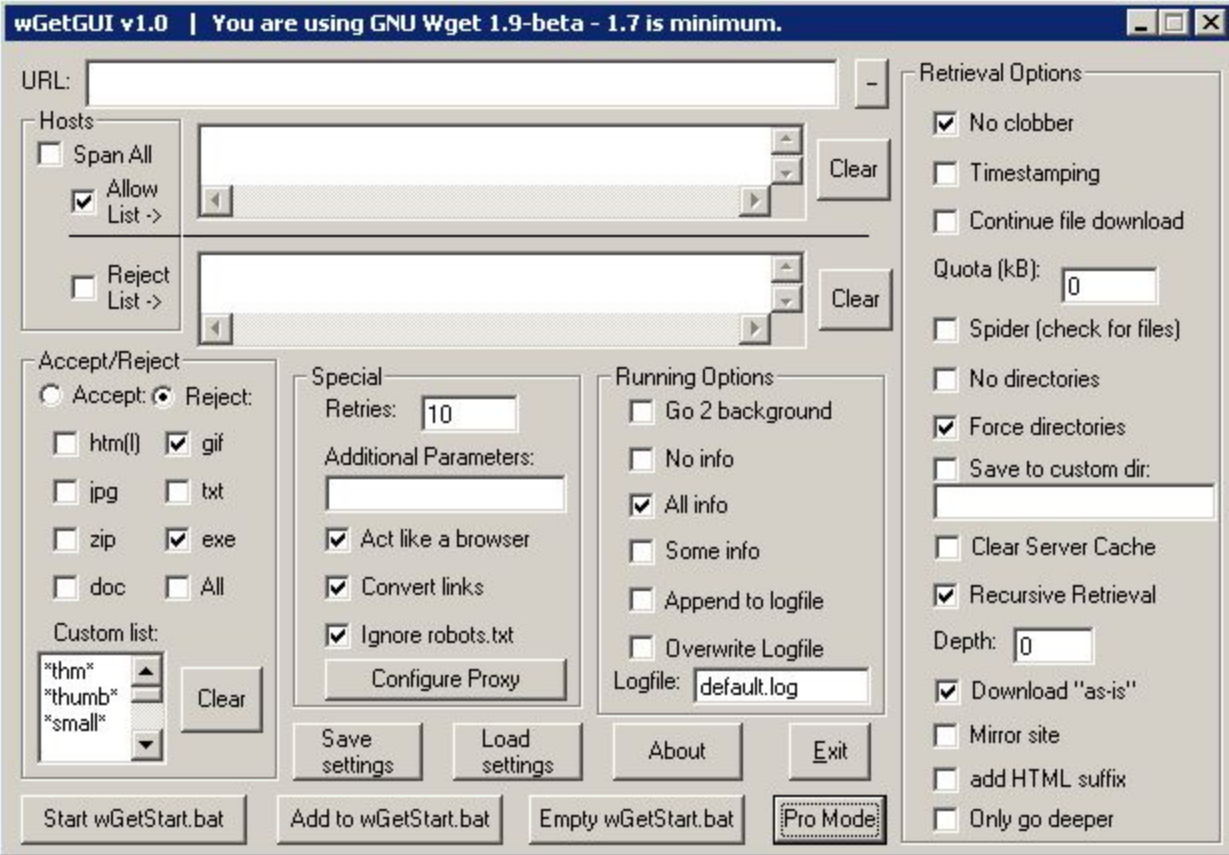


# UI element



Use **visual priority** to  
guide the player's eye





wGet



Go

# Visual Priority

- Players don't immediately read *everything* on a page
- They look at elements in an order based on their **visual priority**
- Zach Gage calls this the **Three Reads**
  - Check out “Building Games that can be understood at a glance” by Zach Gage

# What Affects Visual Priority

- Size
- Movement
- Contrast (not just colour)
  - If your entire UI is yellow, adding more yellow won't make it stand out
  - Contrast with background but also with self
- Shape
  - Jagged shapes over round shapes

OVERWATCH™

1.0.4.2 - 29293



PUDGE77 2

*PLAY*  
*TRAINING*  
*HERO GALLERY*  
*LOOT BOX*

SOCIAL  
CAREER PROFILE  
OPTIONS  
EXIT GAME



ZENYATTA

0/54 UNLOCKS

PRESS ENTER TO CHAT



OVERWATCH

1.0.4.2 2023



PUDGE77



*PLAY*  
*TRAINING*  
*HERO GALLERY*  
*LOOT BOX*

SOCIAL  
CAREER PROFILE  
OPTIONS  
EXIT GAME

PRESS ENTER TO CHAT



ZENYATTA

0/54 UNLOCKS

- PLAY*
- TRAINING*
- HERO GALLERY*
- LOOT BOX*


- SOCIAL
- CAREER PROFILE
- OPTIONS
- EXIT GAME



**ZENYATTA**  
0/54 UNLOCKS



Minimize  
related-information  
distance

 92527

# Ye Olde Shoppe

*I have*



92527

*Total*



82242

# Minimize related-information visual distance

- Distance can be both in both **space** and **time**
- Space is visual distance on-screen
- Time is time between seeing related information
- Put related information nearby, even if it's duplicated

Low on Luxury Food  
Low on Quantum Computers  
Low on Neuro Implants



Temperate			Arctic			Lunar					
30	10	-5	60	15	-31	-2	20	-5	4	30	-9
66	-1	10	44	-66	8	-8	-19	-4	0	0	

Low on Luxury Food  
Low on Quantum Computers  
Low on Neuro Implants



Workers Operators Executives Investors Ornaments





**Electricity**

Electricity Availability

Electricity consumption: 11MW  
Electricity production: 21MW

Legend

Power Plants	Active	Inactive
Buildings	Connected	Disconnected



Minimize  
related-action  
distance

# Minimize Related-Action Distance

- The more often a user performs two actions together, the closer they should be (in distance & actions)
- An action = a click, a button press





49 5



Build



49 5



Production



49 5

 **Skins Dryer**  
A place to dry skins  
4



Left click to place.  
Use Z and C keys to rotate.  
Shift+click to place multiple.



49 5



Build



49 5

Production



49 5

 **Food Dryer**  
A place to dry food  
10



Left click to place.  
Use Z and C keys to rotate.  
Shift+click to place multiple.



Be conscious of  
existing knowledge

# Players are not a blank slate

- Players will come to your game with existing knowledge
  - Culture
  - Language
  - Other games
  - Software
- If your game is at odds with that knowledge, it **makes your game harder to learn**

# Existing knowledge can apply to

- Symbol meanings
- Colour associations
- Input conventions
- Text semantics





Desired Player  
Experience



Game State

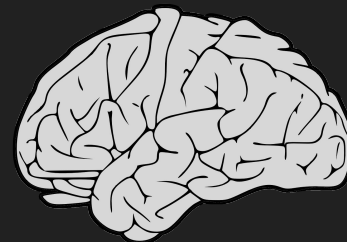


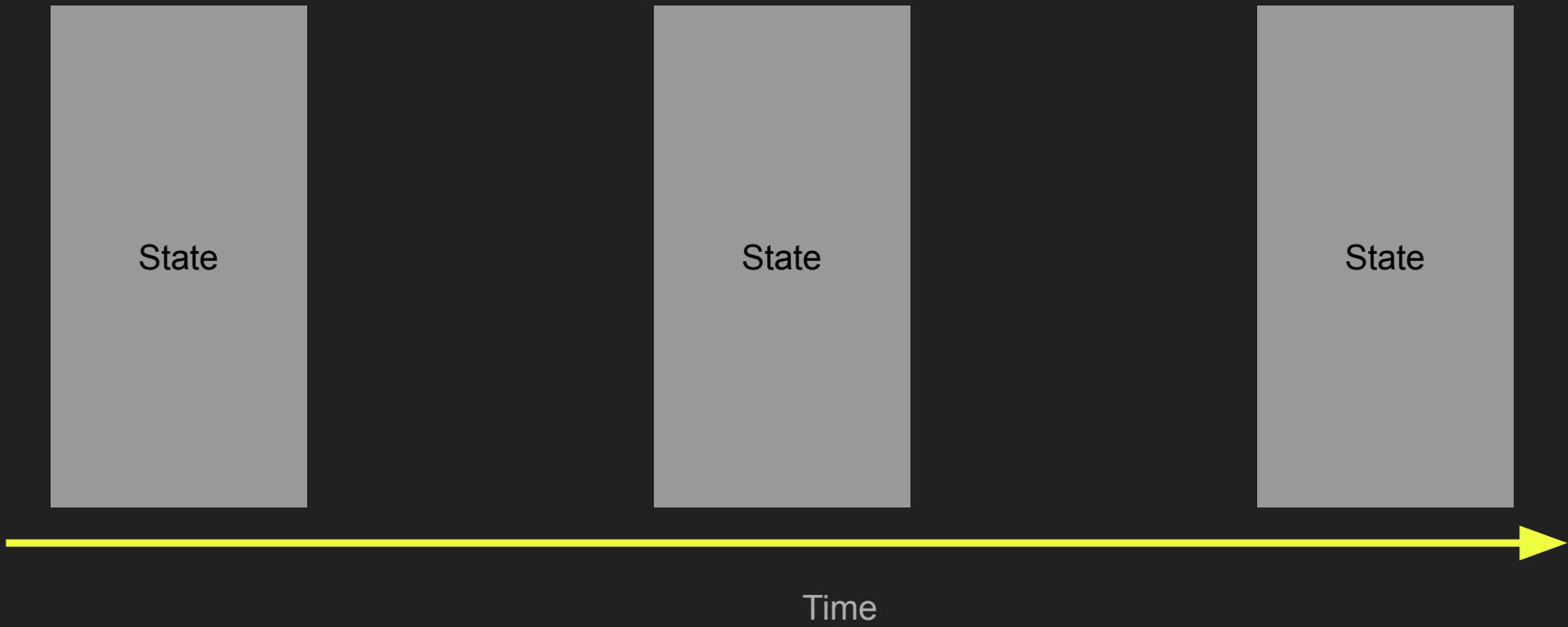
Game  
Mechanics



UI Visuals

Existing  
knowledge





Game State

**Player character**

health = 0.8f;



Time

# State Change

- Game state changes over time
- UI will usually reflect the current state of the game
- Players can't track every single difference





Before

**WORLD TRACKER**

**NUCLEAR FISSION**

Turns: 3 Boost through Great Scientist or Spy.

**SCORCHED EARTH**

Turns: 2 Build 2 Field Cannons.



After



NEXT TURN

# Why state change is important

- Players can be overwhelmed
- Bring player's attention to what's important
- Visual intensity of change proportional to importance

# Low Importance — Mario Coin

- Happens all the time
- Still want juice
- 



# Medium Importance — Mario Power-up

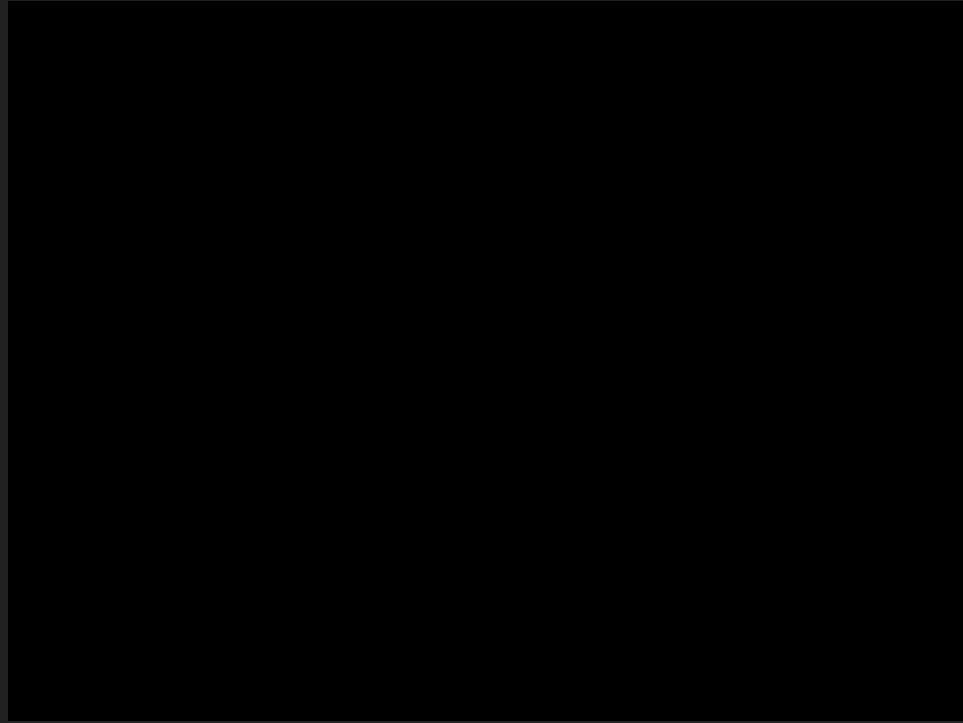
- Happens every 30s-2min
- Every time player picks up a power-up
- **Changes the player's powers**
- Freeze-frame, repeat change state animation
- Sparkles where the item was, making it clear how you touched it



Super Luigi Bros, 25% speed

# High Importance — Overwatch Ultimate

- Gain ultimate ability in Overwatch
- Once every few **minutes**
- Immediate call to action for the player



Overwatch, 25% speed

Use

**Multiple modalities** to  
convey information  
better

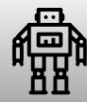
Food

Defense

Horses

Robotics









Food



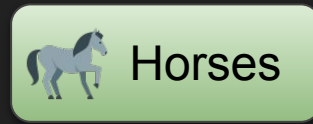
Defense



Horses



Robotics



# Use multiple modalities to convey information

- Text
- Icon
- Colour
- Pick **two**

An Unknown Clan has gained the Fame title **Thanel**



802 AD - April  
We should prepare for winter.

**Plain**  
This area is empty.

2/3 2/2 2/2

2 Farmer 2 Sheep

**Warband**

**Civilians**

Improve Tools

Organize a Feast

**BUILD [B]**

! 3 1 2

# Way more tools out there

- Lenses through which you can analyse/design UI/UX
- Contrast
- Animation
- Affordance

- UX > on-screen stuff
- Tools for UX design
- Check out some games



# My Design Process

- What works for me
  - Change them how you like!
- Design is a **process**
  - You learn by doing it
  - Not just a bunch of rules

# How I Design UIs

1. Pre-production — Research the genre
2. Create mocks of user flow, general layout
3. Create detailed mock-ups of each screen
4. Create in-game “greybox” UI
5. Improve in-game art, animations, interactions

# 1. Research the Genre

# 1. Research the Genre

- Pre-production
  - Continue doing it in production as other new games come out, too!
- Players come to your game with existing ideas
  - From their culture
  - From software in general
  - From other games
- Learn how other games present information
  - Keep bits that work
  - Avoid bits that don't

# 1. Research the Genre

- Find games within the genre
- Play them, **take notes & screenshots**
- Your first 15 minutes is incredibly valuable
  - You're learning as a new user
  - You can never experience the game like that again, you become an experienced user
  - You're doing **user testing** on someone else's game
- How does it feel?
- What is confusing? What is clear?
- What does the game show you? What does it skip?
- What existing knowledge are you using to understand the game?
  - How would someone *without* that knowledge perform?

# 1. Research the Genre

- Pick **medium-quality** examples
  - The best ones are too perfect, too cohesive, seem effortless
  - Worst ones are just all bad
  - Medium ones show you what it's like to **lack certain elements**, this is educational!

# What does research look like?

Some **positive points** on design, feelings when using/playing:

Clean, minimal UI with Nordic flourish

Text uses icons & colours to help comprehension

Consistent button/icon/text size

Buttons feel chunky in a good way, animation + sound

# What does research look like?

Some **criticism**:

Zero feedback on giving orders, not clear if anything happened

Cursor only has two states, normal/shiny

Tooltip text all same size, no clear **hierarchy** of importance

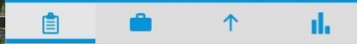
Right-click gives orders, but this is **never explained**

Nothing is satisfying to click on





JOHN AGUILAR



STAFF INFORMATION

Janitor status  
**Going to clean litter**

Workload  
**Normal**

Trainee 0	0
0	0
0	0
0	

RECENT THOUGHTS

- Everything is all right I suppose
- I'm content with my paycheck

STAFF HAPPINESS



## 2. User flow

# Minute, hour, day guideline

- Changes become way more expensive later in the process.

Paper sketch

1 minute

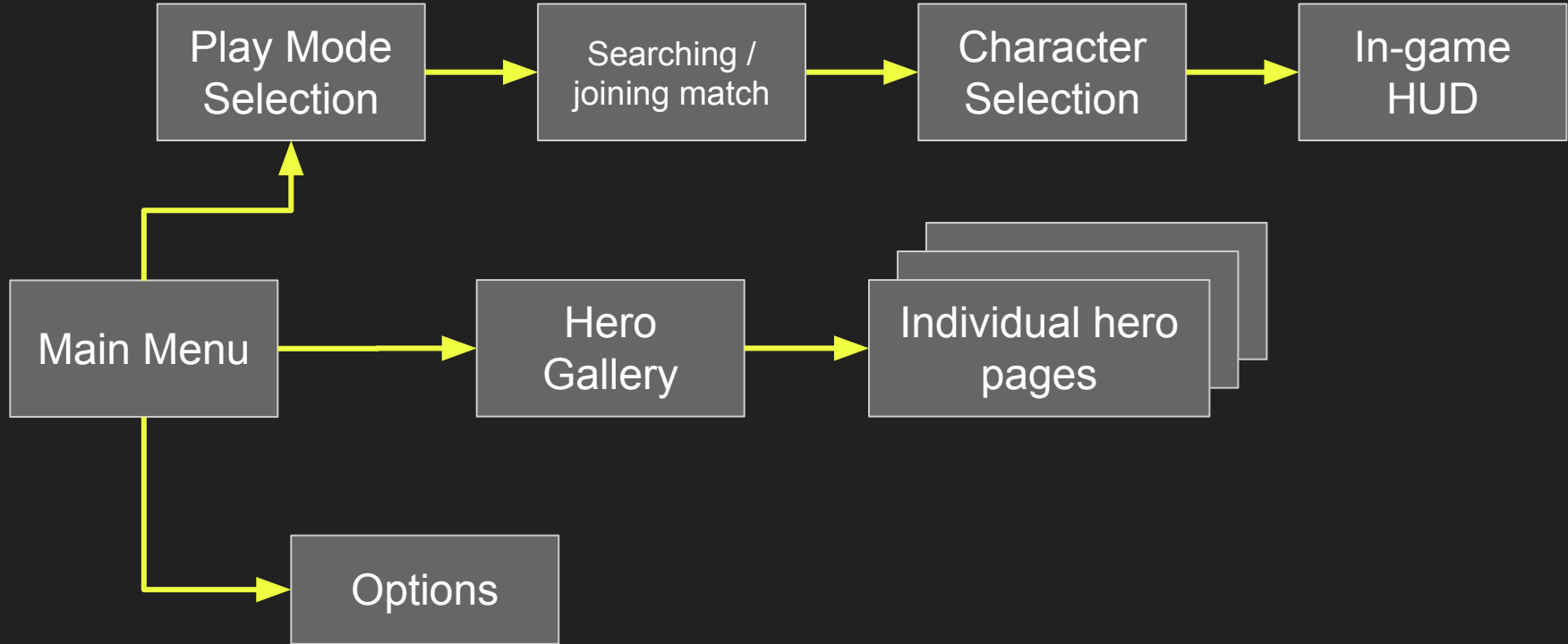
Styled mock

1 hour

Code + assets

1 day

# Overwatch Menu Graph



### 3. Per-element design

## 3. Per-element design

- Recursively design from largest to smallest
  - Start with pages
  - Then large elements like windows
  - Then elements within there
- List things to communicate, actions to perform
- Order them by importance
- Try different layouts

Navigation  
Items

Character

**ZENYATTA**  
0/54 UNLOCKS

Selected Character

Current team composition

Character list



# Game Title

Menu Options

Company  
Links

Status Bar

Notifications Menu

Building Selection

Mini Map

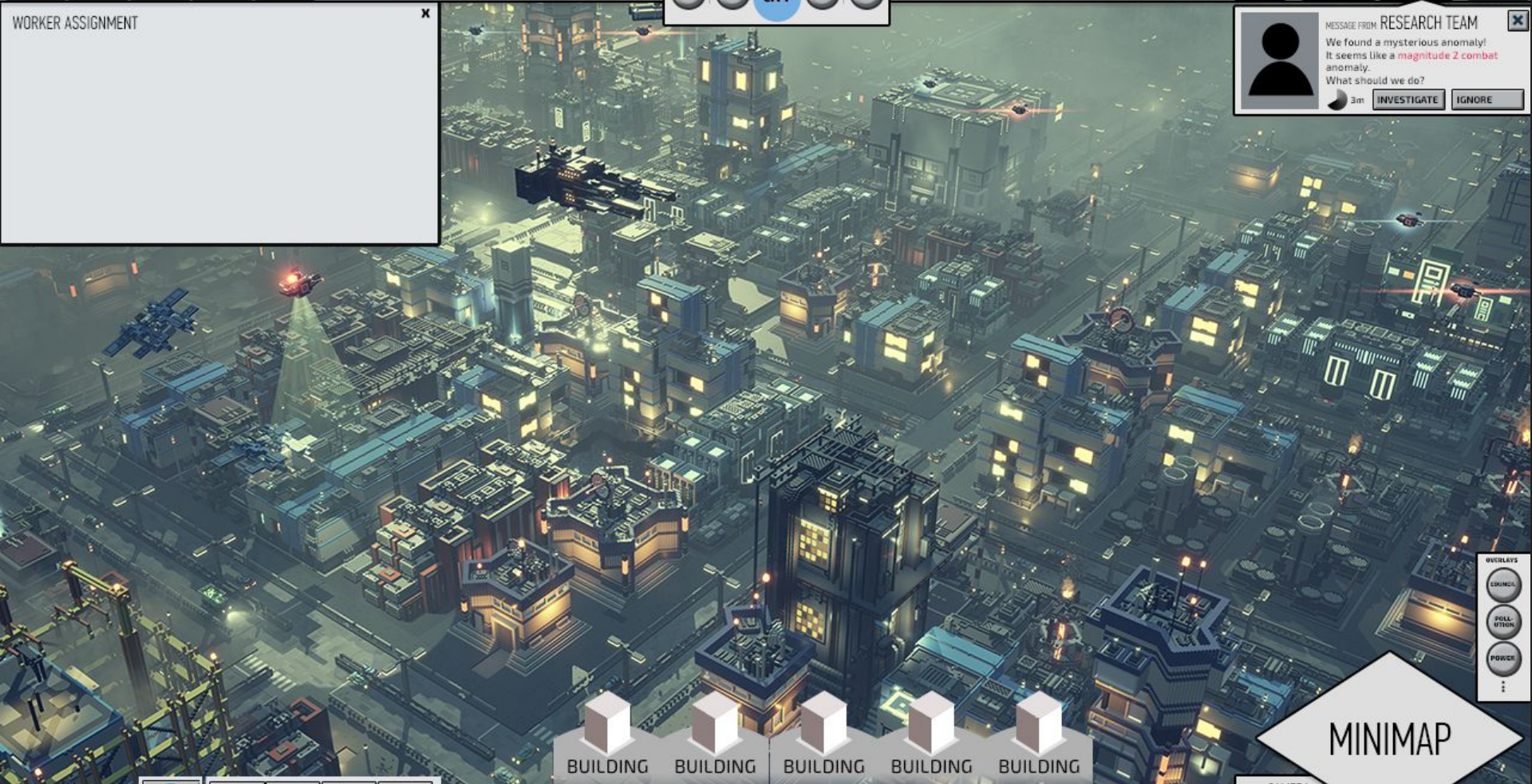
Based on this...

WORKER ASSIGNMENT

MESSAGE FROM RESEARCH TEAM

We found a mysterious anomaly!  
It seems like a **magnitude 2 combat** anomaly.  
What should we do?

3m INVESTIGATE IGNORE



BUILDING BUILDING BUILDING BUILDING BUILDING

BUY TILE SURVEY RESEARCH EXCAVATE DESTROY

Tile Actions

CATEGORY CATEGORY CATEGORY CATEGORY CATEGORY CATEGORY CATEGORY

MINIMAP

CAMERA CONTROLS

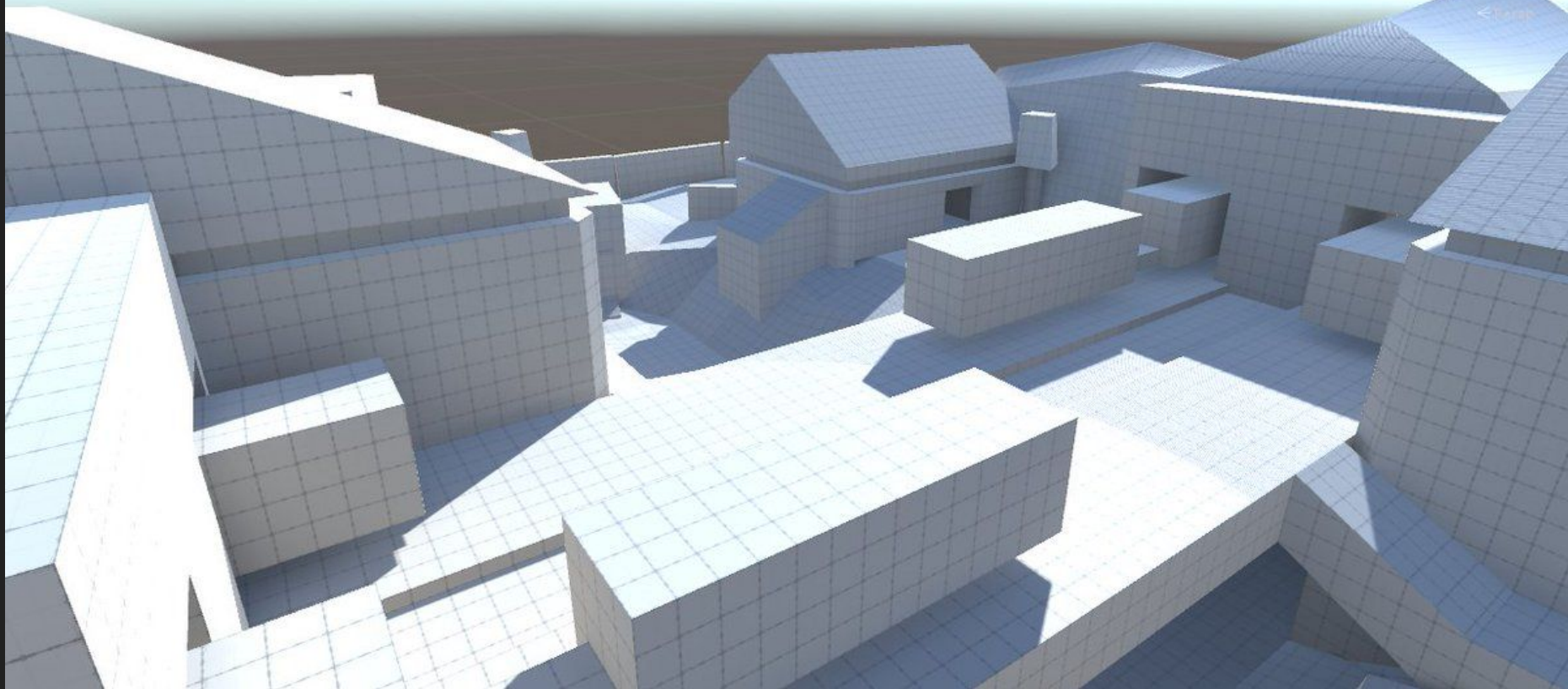
OVERLAYS

COUNCIL POLLUTION POWER

## 4. In-app greybox



< Home



## 4. In-app greybox

- Aka wireframe
- All functionality, minimal aesthetics
- Element size, position are the priority
- Enough visuals to convey states
  - Disabled button state
  - Text colour to highlight key points
- Keep it simple, fast to iterate on
- By getting it in-game you will discover many problems with your designs

5. Improve ingame art



## 5. Improve ingame art

- An entire topic unto itself
- Not something I'm an expert on
- Easy to find ideas online

# Iterating on Interactions

Refresh to get new data

- Button style
- Drag-to-refresh



by Tamara R for Tubik

<https://dribbble.com/shots/2101933-GIF-for-Pull-Down-Space-Ship>

# Summary

- UIs do more than just showing game state to the player
- UX tools can be useful

Thanks!

*Any questions?*

# What helps me

- Pen & paper, whiteboard
- SketchApp
- Coworkers
- Playing a lot of games
- Using a lot of software



# 6. Improve Individual Interactions

(optional)

# Example: “Are you sure?” Confirmation Panel

- Want to avoid users accidentally performing destructive actions
  - Deletion
  - Unsubscription
- Many ways to achieve this
- Be adventurous!



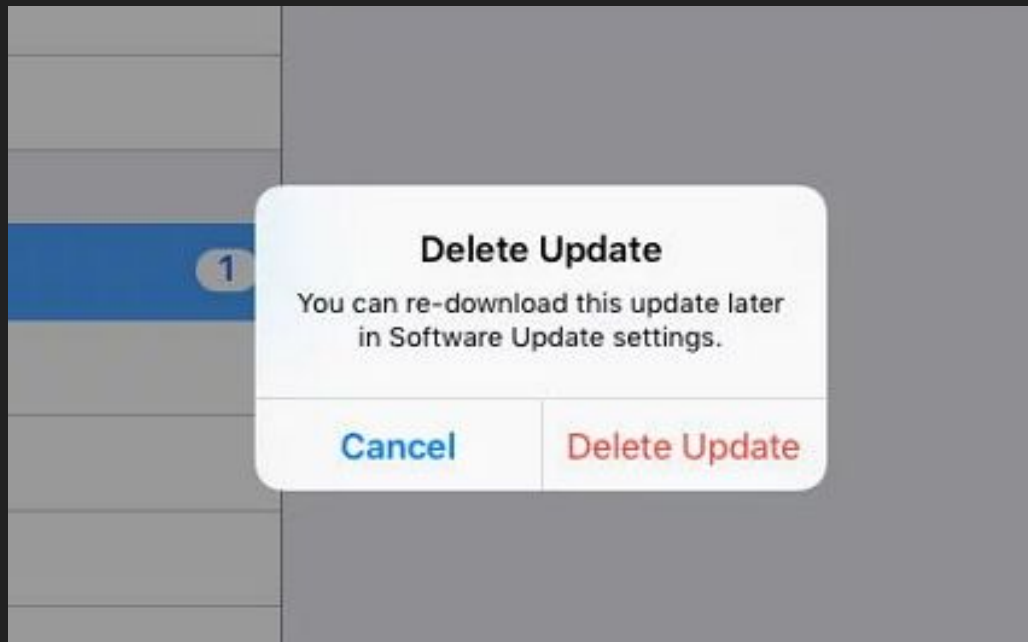
# “Are you sure?” — ~~Lazy~~ Efficient Option

- Don't do anything special
- Some people will make a mistake
- Eh, it's fine
- Just have a regular button
  - Maybe make it red

**Delete Universe**

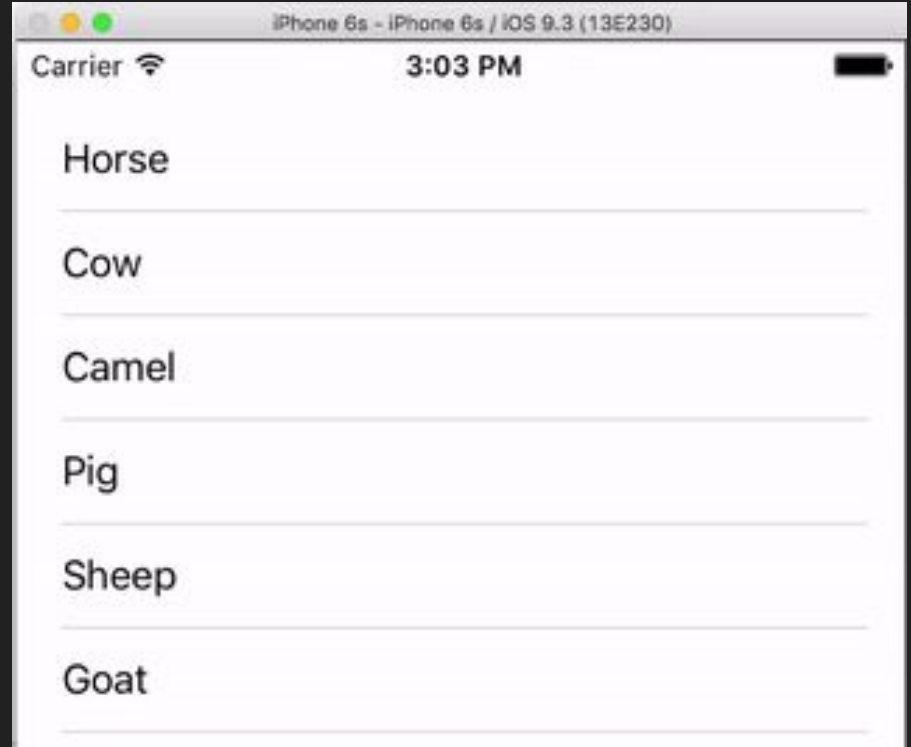
# “Are you sure?” — Modal Window

- Simple-ish
- Blocks interaction
  - Interrupts the user
  - Can be annoying
- Can have explanatory text!



# “Are you sure?” — Swipe

- Only works on touch
- Hidden functionality
  - Low discoverability
- No space for explanation
  - Only single button + short text label
- Only works in lists



# “Are you sure?” — Long-press

- Controller input
- Not great for accessibility
  - Some users can't long-press



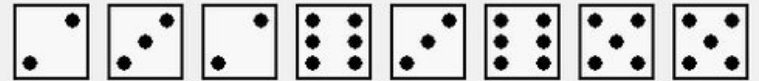
# Don't go too weird...

Volume: 50

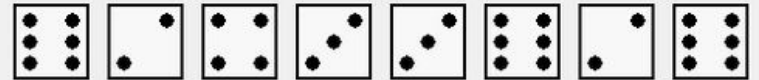


**Current Volume Level: 0%**

Change



Hold  Hold  Hold  Hold  Hold  Hold  Hold  Hold



Hold  Hold  Hold  Hold  Hold  Hold  Hold  Hold

Roll

Volume: 64



# Industry Tips

#1

50% of your work may get thrown away

Specifications change all the time

It sucks, but try not to get too disheartened

# Industry Tips

#2

When you interview at a place, it's a two-way process

You're trying to find out if the place suits you too

# Industry Tips

#3

The “genius difficult programmer”  
myth is toxic AF

Game development is  
90% collaboration

Learn to work with people,  
take criticism,  
give constructive feedback,  
explain your ideas to people