

Feature Specification – User Interface

1. Dash cooldown

1.1 Functionality

Dash cooldown icon is only shown when the dash was used, hidden by default

Circular progress “bar” visualizes the time it takes until the ability is ready again

Starts at the top center and progresses clockwise

Current value of progress bar = $(\text{timeElapsedSinceDashUse} / \text{dashCooldown}).\text{Clamped}(0,1)$

Once cooldown elapsed, dash cooldown icon is hidden again

1.2 Audiovisual requirements

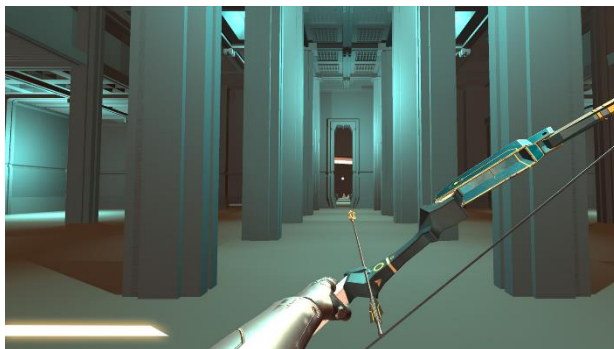
Positioned in the lower right corner

Does not take up much screen space

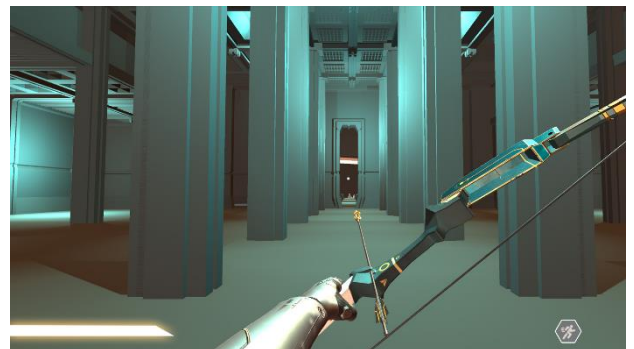
Icon should communicate it is related to dash ability

Flat color overlay is used for the circular progress “bar”

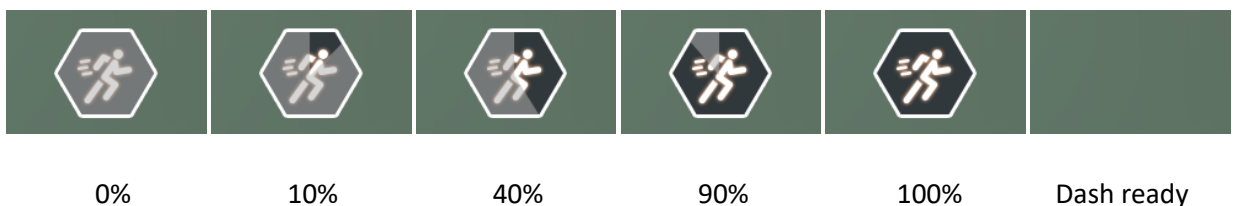
1.3 Example



Invisible when dash is ready to be used



Appears in lower right corner when on cd



0%

10%

40%

90%

100%

Dash ready

1.4 Ideas for future stages

Icon could animate somehow, giving feedback if player tries to use dash while it is on cooldown

Icon could fade out, stressing that the dash is ready again

2. Crosshair

2.1 Functionality

Crosshair has two different modes, exploration and combat

Always starts in exploration mode

Goes into combat mode when pressing left mouse button (to shoot)

Returns to exploration mode when player does **not** shoot for some time

Timer starts once you fire the first arrow and is reset every time you shoot another one (always starts/resets on releasing the arrow)

While you are drawing the bow, the timer is paused (combat mode **cannot** be left while drawing the bow)

If timer reaches a certain value (which should be an exposed variable), the crosshair transitions back to exploration mode and the timer is stopped

Exploration mode has **no** special functionality

In combat mode, parts of the crosshair (that are **not** visible in exploration mode) move inwards to resemble the progress of drawing the bow

The duration of this animation depends on the value of the maximum draw time variable

An orange glow effect signifies when you have reached the maximum bow tension

2.2 Audiovisual aesthetic

Positioned in the very center of the screen

Contrasts well with environment and enemies

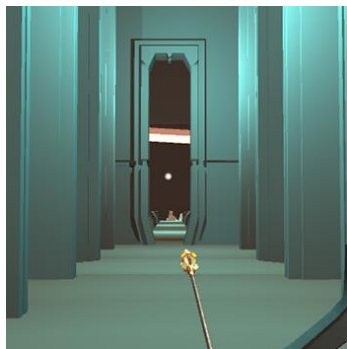
Exploration mode has a simple dot crosshair so players can focus on environment and DevNotes

Both crosshair modes do not take up too much of the screen space

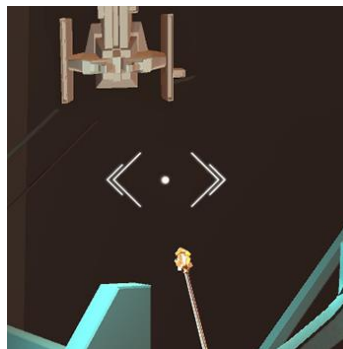
The additional visual elements of the combat mode move inwards to resemble drawing the bow

An orange glow on the crosshair signifies when the bow has reached maximum tension

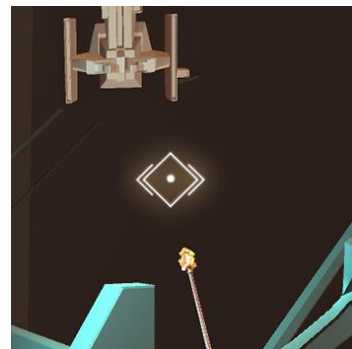
2.3 Visual concept



Exploration mode



Combat mode



Combat mode – bow fully drawn

2.4 Ideas for future stages

Crosshair could change its appearance on killing an enemy

Visual effect could give feedback on releasing the arrow

3. Enemy health bar

3.1 Functionality

Represents the (percentage of) health an enemy has left

Updates every time the enemy loses or gains health points

Health bar is attached to enemy in space

Always rotates to face player camera

Appears when enemy receives damage

Visual effect provides feedback when enemies receive damage

Positioned at the new health value on the bar

Vanishes when enemy resets (and thus regenerates health) or when enemy dies

3.2 Audiovisual aesthetic

Positioned in world space, above the model of the corresponding enemy

Black bar background contrasts well with the environment and the white fill color

Bar dimensions roughly match the dimensions of enemies horizontally

4-5 health bars can be visible at the same time without cluttering the screen too much

Effect provides feedback when enemies receive damage and highlights the new health value

3.3 Visual concept



Enemy not yet attacked



Enemy with full health



Damaged enemy

3.4 Ideas for future stages

Amount of dealt damage could be highlighted on the bar itself

Could add visual elements to indicate the absolute amount of health

4. Player health bar

4.1 Functionality

Represent the health points of the player

Updates every time the player loses health points, regenerates health points, or respawns

Only appears when player is in combat **or** regenerating health

Hidden by default

Turns visible once player is detected by an enemy

Either by entering their cone of vision or by hitting them with an arrow

Is hidden again when player **neither** deals **nor** receives damage for some time

Timer starts once health bar is shown and resets every time the player deals or receives damage

If timer reaches a certain value, the player exits combat mode

If **no** regeneration timer is running at this point, the health bar disappears

The duration (in seconds) to exit combat mode should be an exposed float variable on the Health and Respawn script of the player game object

Turns visible when player uses their regeneration ability

After using the ability, a timer (independent from the combat timer) starts to run

If timer reaches a certain value **and no** combat timer is running, the health bar disappears

The duration (in seconds) should be an exposed float variable on the Health and Respawn script of the player game object

Important: The regeneration and combat timer work together to hide the health bar

Each of them checks on expiry whether the other timer is still running

If so, the bar remains visible

If **not**, the bar is hidden

Visual effect provides feedback when player receive damage

Positioned at the new health value on the bar

4.2 Audiovisual aesthetic

Positioned in the lower left part of the screen

Does not take up too much screen space vertically

White orange glow contrasts well from dark blue environment and draws a lot of attention to it

Is hidden when not in combat so players can focus on the environment and DevNotes

Visual effect indicates when player takes damage and highlights the new health value

4.3 Visual concept



Hidden health bar while not in combat



Health bar during combat



Health bar and effect when taking damage

4.4 Ideas for future stages

Bar could be animated instead of jumping to the next value immediately

5. Screen effects

5.1 Functionality

No screen effect by default

Player enters critical state when health points drop below 30% of the maximum amount

Red screen effect signifies this state

Only visible while player is in critical state **and** in combat

If the health bar is **not** visible, the screen effect will **not** be visible either

Different screen effect is shown when player regenerates health

Blue screen effect signifies this state

Visible in **and** out of combat, as long as you regenerate health

Regeneration ability will **not** be part of the second stage, so there is **no** way to trigger this effect right now

Not shown if player is already at full health and tries to regenerate

A mere reset of health points (e.g. due to respawning) does **not** trigger the effect

Has priority over critical state screen effect

I.e. if player is below 30% health and tries to regenerate, the blue screen effect replaces the red one

5.2 Audiovisual aesthetic

Low opacity lets player focus on gameplay

Critical state uses red color

Signifies danger

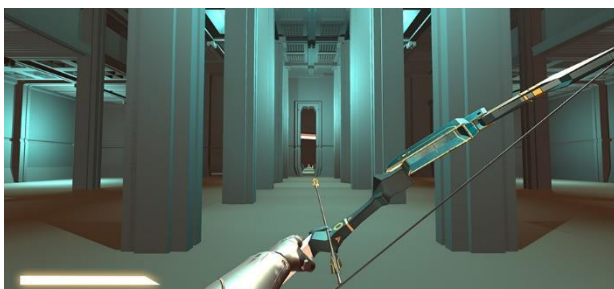
Contrasts really well from rest of the game environment

Regeneration state uses blue color

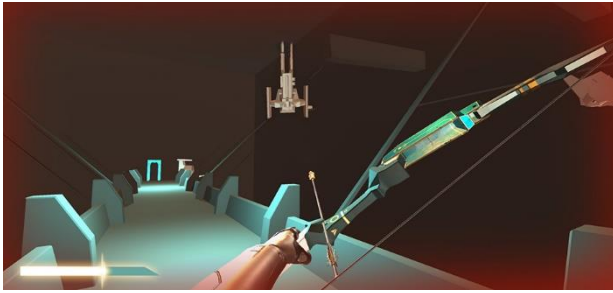
Draw less attention to it because of the overall color of the environment

Ability is related to the god of patience

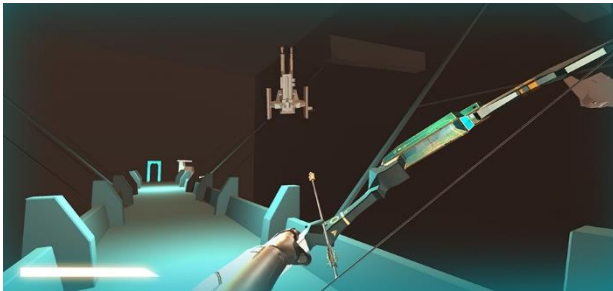
5.3 Visual concept



Default screen, no effects



Critical health state, red screen border



Regeneration state, blue screen border

5.4 Ideas for future stages

Critical health effect could be animated in a pulsating way to express urgency

Might need a subtle screen effect for when player takes damage