**WildHacks Architecture Document**

**Members:**

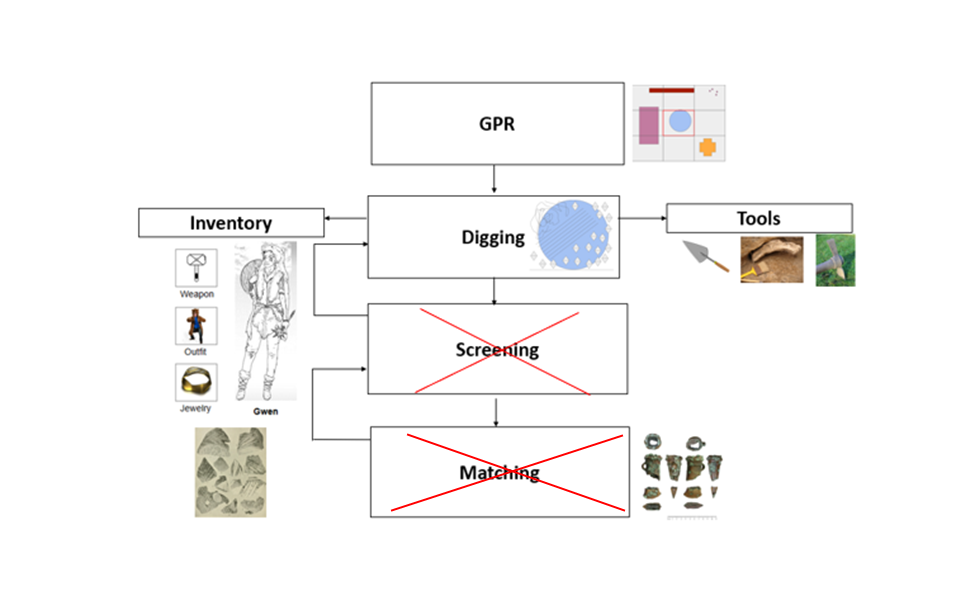
Nathan Church: Testing: Word Count: 395

Lee Bills: Detailed Design (GPR, Digging, Tools, Metrics, Quality Review Writeup): Word Count: 519

Madisyn McFarlane: Detailed Design (Screening, Matching, Inventory): Word Count: 424

Ruth Dankwah: High Level Design (Architecture): Word Count: 219

**High Level Design (Architectural Diagram)**



* **GPR:** Allow players to ping the specific area in the ground to identify and retrieve artifacts. Players will begin with a grid screen with a top-down view of the digsite.
* **Digging:** Players will dig through different soils, represented by different textures (Scribble: fine soil, Diagonal: medium soil, Diamond: rocky soil) to retrieve inventory items. ~~Soils will change as the player digs deeper.~~
* **~~Screening:~~** ~~After digging, players should discover artifacts. Players must use tools such as a brush on artifact to gain a clear sight on what it is. There is an arrow going from screening to digging to represent the loop that should continuously occur during the digging and screening stage.~~
* **~~Matching:~~** ~~Players should try to piece the broken artifacts together such as a puzzle. The artifacts found should reproduce the original artifact. There is also another loop from matching to screening to represent the cycle that occurs when cleaning the artifact and seeing what it is, to piecing it together.~~
* **Inventory:** Inventory should ~~display~~ record all the items found by the player, as well as items needed to be found as the game progresses. Retrieval of inventory items allows players to gain points throughout the game.
* **Tools:** There will be tools ~~and weapons~~ provided to the players during the game.

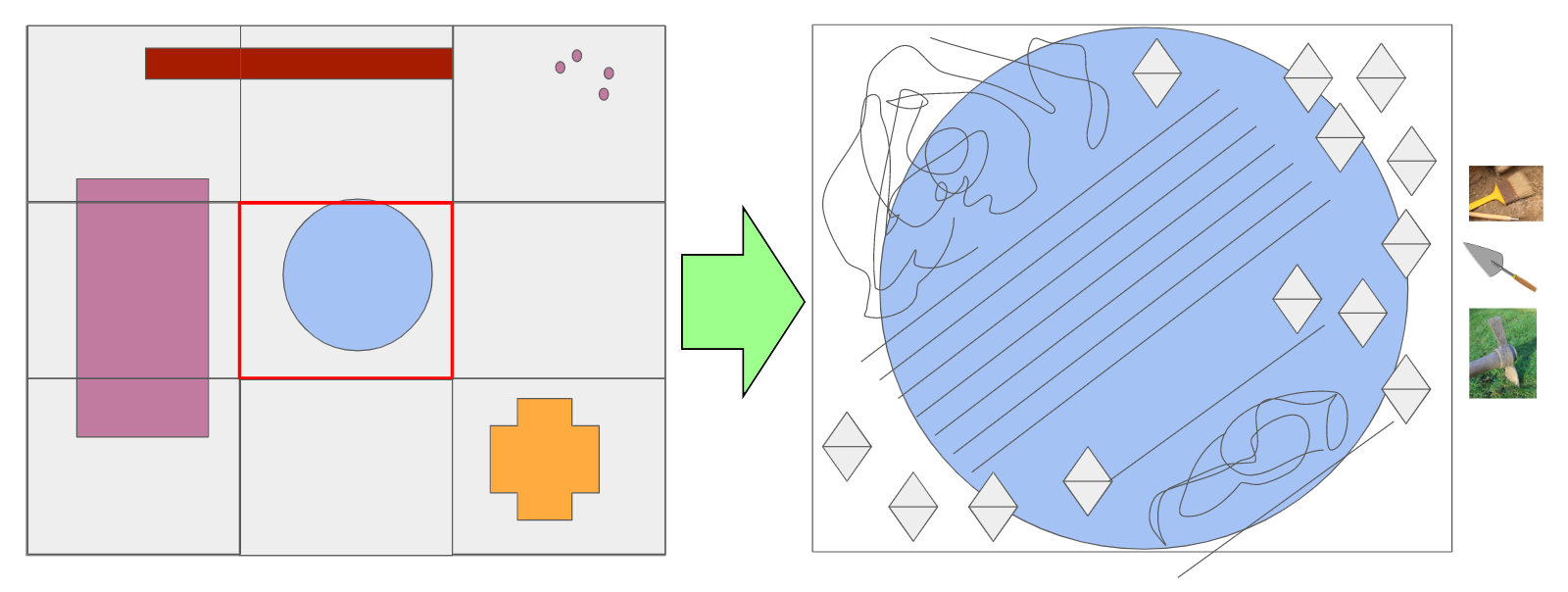
**Detailed Design**

The division of our high-level design into methods, which when implemented will be one Unity script per method

* GPR
  + When a grid square is selected with the radar, highlight ~~outlines of~~ structures present under the soil there, if any
  + After using the GPR, the game will zoom in to focus on the selected grid square
  + Ensure that GPR is only available while in the zoomed-out view. Note that the zoom functionality has been added to a new tool, the magnifying glass, for the player to manually activate
* Digging
  + Upon starting the puzzle, all grid squares are seeded with artifact pieces, soil types/layout, ~~and structures~~. Certain routines will be used to seed the map, according to structures and dig site location
  + After the GPR and magnifier are used and the game zooms in, draw soil distribution
  + When the player uses a digging tool on a soil patch, determine if the swing was successful. ~~Additionally, compare the tool to the soil, as well as artifact presence, to see if that artifact is damaged~~
  + ~~The first time an artifact is damaged, display a tooltip informing the player of their mistake~~
  + ~~If soil was successfully removed, send it to the Screener for processing and unearth new layer of soil, if there is another underneath~~
* Tools
  + ~~Upon starting the puzzle, initialize all tool performance modifiers according to the player’s upgrades~~
  + When the HUD element to switch tools is activated, swap the player’s active tool to the new tool
  + Constantly update the tool HUD to reflect the tools available at that stage of the puzzle (GPR vs Digging vs Matching)
  + Currently selected tool should be highlighted
  + Keep track of which tool is currently active
* ~~Screening~~
  + ~~Soil will automatically be populated into the sieve located in the bottom corner of the interface after a successful removal of soil during the digging phase~~
  + ~~The sieve will play a shaking animation, and the soil will be removed from the sieve~~
  + ~~If there was a small artifact part located in the soil, the part will be left in the sieve after the soil is removed, play a short glowing animation to get the player’s attention, and then it will be moved to the progress inventory~~
  + ~~If there was not a small artifact part located in the soil, the sieve will be emptied of soil, and no other actions will be taken~~
* ~~Matching~~
  + ~~Once all parts of a particular artifact have been collected in the progress inventory, the player will receive a message on the screen telling them so, and then the matching interface will automatically interrupt the digging interface~~
  + ~~The largest part of the artifact will be shown in the middle of the screen, and the player can click and drag the smaller parts from the part inventory on the side of the interface to fit them into the missing sections of the artifact~~
  + ~~If a part is dragged and let go on the incorrect area of the puzzle, an ‘incorrect’ audio will be played and the part will appear back in the part inventory~~
  + ~~If a part is dragged and let go on the correct area of the puzzle, a ‘correct’ audio will be played and the part will be integrated into the artifact~~
  + ~~Once all parts are placed correctly into the puzzle, a glowing animation will be played on the completed artifact, as well as a ‘success’ audio, and the matching interface will change back to the digging/screening interface. The completed artifact will then appear in the progress inventory~~
* Inventory
  + ~~The progress inventory will be located on the side of the GPR and digging interfaces~~
  + Upon the initial loading of the game, the progress inventory will include only a count of ~~how many objects are hidden in the puzzle,~~ how many of these objects have been found and completed so far (Upon first play of a level, this number should be 0), ~~and the amount of grant money the player currently has (This will be input by the full game when the puzzle game loads)~~
  + ~~As artifact parts are found either in the digging or screening process of the game, they will appear in the progress inventory beside similar parts, which will eventually fit together to create a completed artifact in the matching process of the game~~
  + ~~After the matching process is successfully completed, the individual parts of the artifact completed will disappear from the inventory, and the complete artifact will appear alongside the name of the artifact. The completed artifact count will be updated accordingly~~
  + ~~The grant money count will be updated accordingly as well, with artifacts that were broken during the digging process worth less money than artifacts which were unearthed with the correct tools~~
  + Once the player finishes the level, the list of complete artifacts currently in the inventory and the amount of grant money the player earned during the level ~~will be output back into the full game~~ Currently, it just keeps track internally.

HUD Diagram:

Left: Wide View, Right: Zoomed in, Bottom: Matching Interface



**Inventory**

Artifacts: 0/12

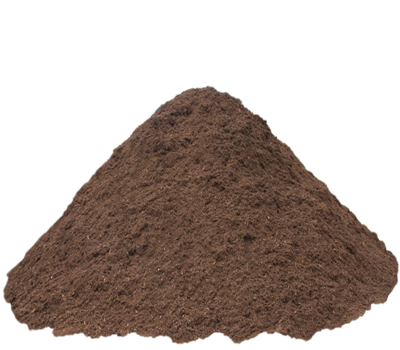
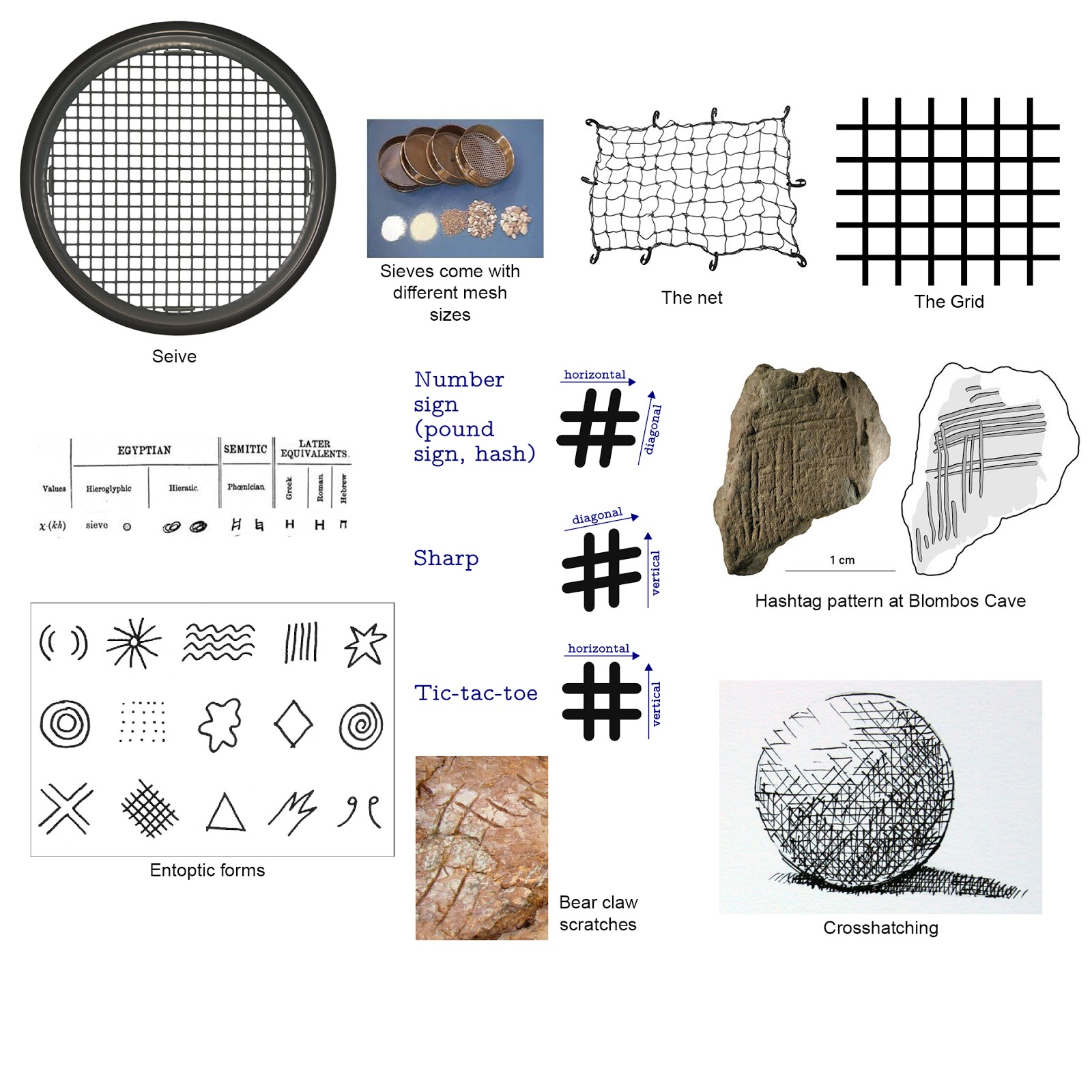
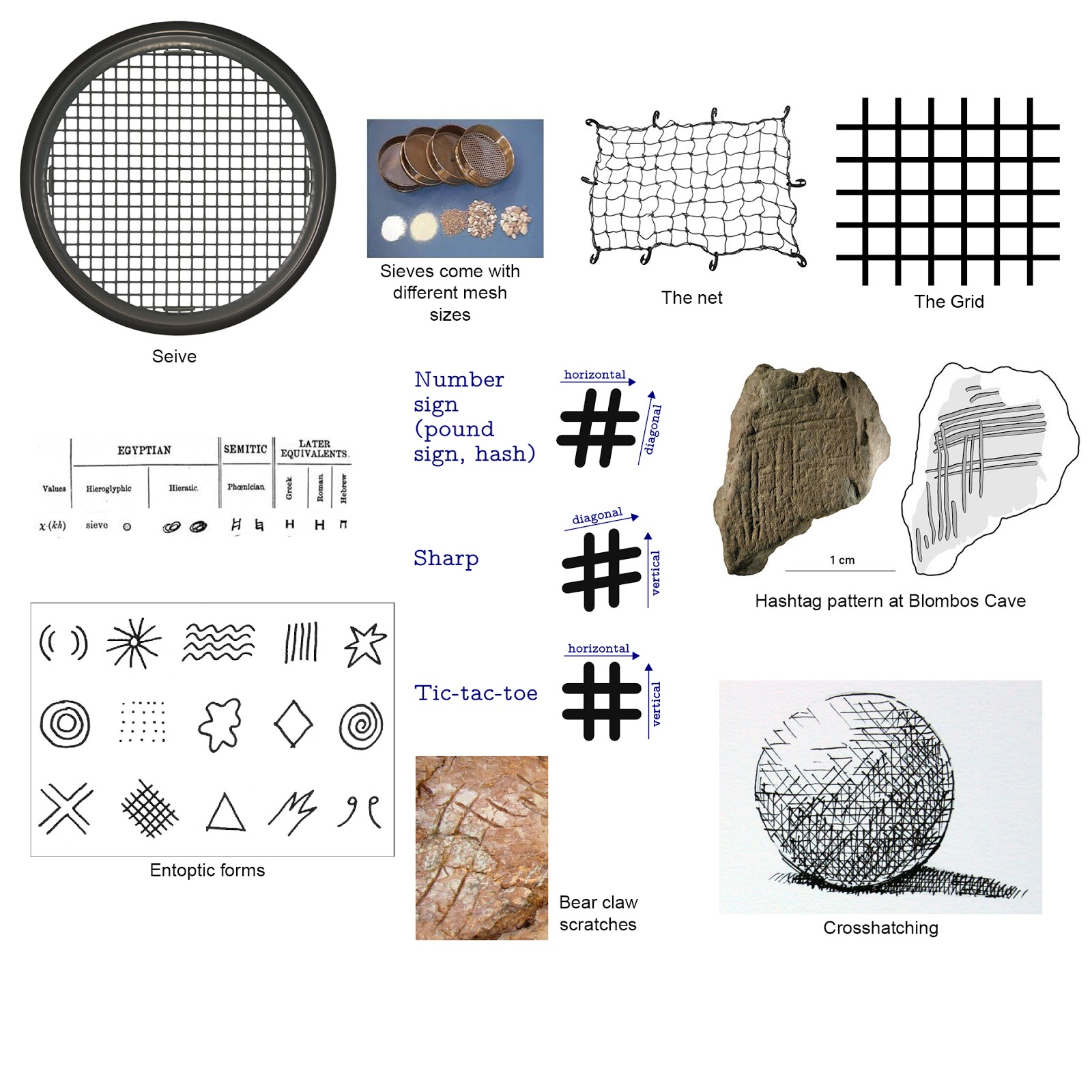
$125

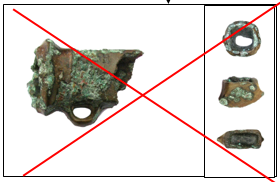


**Inventory**

Artifacts: 0/12

$125





**Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Case #** | **Test Scenario** | **Test Steps** | **Test Conditions** | **Expected Results** |
| 1. | Selecting a grid square using the GPR | 1. The player selects a digsite location 2. Select a grid square with the radar | The player has access to the digsite and the GPR | 1. ~~Highlight the outlines of any structures~~ Show an icon if something’s there under the soil 2. Zoom the play area into the selected grid |
| 2. | Drawing the soil distribution of a grid area | 1. The player selects a digsite location 2. Select an grid area with the ~~radar~~ magnifier | Routines to seed the grid squares with artifact pieces, soil types/layout, and structures depending on structures and dig location | 1. The square is zoomed in on. 2. The game draws the soil distribution of the selected grid area |
| 3. | When the digsite screen is zoomed out, only the GPR and magnifier is available to use. | 1. The player selects a digsite location | The player is at a digsite | The only tool available in the zoomed out view is the GPR and magnifier |
| 4. | Determine the result of using a digging tool on a soil patch | 1. Select a digsite location 2. Select a grid square 3. Select a digging tool 4. Select a patch of soil | ~~All digging tools’ performance modifiers according to the player’s upgrades.~~  Artifact locations within the grid square. | Determine if the digging was successful, ~~if it did succeed, send the soil to the Screener.~~ If there was an artifact in the selected soil patch, compare to the selected tool, ~~and determine if the artifact was damaged or not. If an artifact was damaged for the first time, display a tooltip.~~ |
| 5. | Swapping the player’s active tool | 1. The player selects a digsite location 2. Select a grid square 3. Select the HUD element to switch tools | The player has more than one digging tool | The player’s active tool is changed. |
| 6. | Discovering small artifacts within soil patches | 1. Select a digsite location 2. Select a grid square 3. Select a soil patch with the active tool | There is a small artifact in the soil patch.  The digging of the soil patch is successful. | ~~The sieve plays a shaking animation.~~  A ~~small glowing~~ bouncing animation is played.  The artifact is moved to the player’s progress inventory. |
| ~~7.~~ | ~~Assembling a piece into the wrong location~~ | 1. ~~Drag an artifact piece to the wrong location~~ | ~~All parts of an artifact have been found and are in the progress inventory.~~  ~~The player is taken to the Matching screen.~~ | ~~An incorrect sound effect is played.~~  ~~The dragged part is put back into the part inventory.~~ |
| 8. | ~~Assembling a piece into the right location~~ | 1. ~~Drag an artifact piece to the right location~~ | ~~All parts of an artifact have been found and are in the progress inventory.~~  ~~The player is taken to the Matching screen.~~ | ~~An correct sound effect is played.~~  ~~The dragged part is integrated into the artifact.~~ |
| 9. | ~~Assembling an artifact in the Matching screen~~ | 1. ~~Drag each artifact piece into their correct locations~~ | ~~The player has all the artifact pieces for the artifact they are matching pieces of.~~ | ~~The artifact plays a glowing animation along with a success sound effect.~~  ~~All the artifact’s pieces are removed from the progress inventory.~~  ~~The completed artifact is placed into the progress inventory.~~  ~~The game moves back to the digging/screening interface.~~ |

**Quality Review**

* Functionality - Our plan for the puzzle is structured as to enable seamless, secure integration with the rest of the game’s framework. Additionally, it meets all functional requirements.
* Reliability - Our puzzle design is robust, and leaves little margin for error (erroneous results, etc) in all its moving parts. The puzzle experience will be consistent across playthroughs.
* Usability - The user interface, as designed, is intuitive, and will promote an engaging and fun user experience
* Efficiency - The high-level method design produces a minimum amount of overhead, and the requirements have been consolidated to the simplest structure. Unity will handle memory and resource management, enabling lightweight performance.
* Maintainability - The architecture of our puzzle is intuitive and modular, allowing for quick modifications. This is achieved through a fine division of methods into scripts.
* Portability - Unity’s Rewired plugin will enable our puzzle’s compatibility on all planned platforms, even accounting for different input devices.

**Metrics**

* Product size: ~~roughly~~ **~~28 user stories~~** ~~(corresponds with described methods in Detailed Design)~~ 20 User Stories
* Product Complexity: **814.14 function points (**using a Function Point Analysis calculator)
* Defects: **none currently detected,** as our project is part of a large video game that has been in development for a while. The framework and structure have already been established and appear to be effective in the game’s development and operation.

**Developer Notebook**

<https://www.evernote.com/pub/madisyn101/cs499developernotebook>