ThinkItThru!

### By Jensen Schmidt, Levi Wright, and Zachary Travis

## Requirements

## Design Description

### Outline (this is just for us)

* Overview (our pitch)
* Feature list / Site Layout (basically an intro to what we’re covering, move into UI storyboard)
* Functionality (deeper dive into features, big section where I’ll reference component diagram
* Server Stuff (reference block diagram and ERD diagram, good luck Jenson)

Presentation outline:

* Overview/Introduction (Z)
* Block Diagram (J)
* Server stuff (J)
* Component diagram and Feature List (functionality) (L)
* Storyboard (Z)

### Overview

ThinkItThru! will be a gamification and organization tool designed to help users organize tasks and motivate them to complete all of the work they have. The website's goal will be to encourage time spent on tasks and an increase in self-assignment of projects and longer-term goals. Users will accomplish this by creating Tasks for themselves, which will be scheduled into goals. Users will then be rewarded for working on and completing these Tasks.

### Features and Layout

(Figure 3) Users will be greeted by a sign in page where they will be able to sign in with Facebook, Google, or X. If they are a new user, they will be directed to a registration page. This page will ask for the user’s name, their major, and ask them to upload a profile picture. The user will be sent to the Dashboard page, where Task Data is held. Users may add Tasks here, which will be used to create daily and weekly objectives for the user to complete to stay on course. Users will “clock in” and “clock out” of a task to record time spent on it. Spending time on a Task, marking a Task as complete, and marking all daily or all weekly tasks as complete will cause the user to gain an “XP” reward.

From the Dashboard screen, users can use the navigation bar to visit more pages (Figure 3). The Profile screen allows the user to edit their profile, such as their name and picture. The Gameboard and Inventory screen allows users to purchase items, hold them in their inventory, and place them on a garden-themed game board to display their items as well as receive an “XP” bonus for their work. Pricier items and items that use up more space in the garden will have a greater reward multiplier.

Another page accessible from the Navigation Bar is the User Log page. This page keeps track of user activity with two graphs. The first is the Daily Graph, which shows a day-by-day breakdown of work recently done in the past 7 days. This graph is specific in showing what specific tasks were done in each day. The other graph is the Weekly Graph, which shows the amount of time worked during each week of the past year (52 weeks). This chart is meant to show how, over time, a user’s activity time has changed- hopefully for the better.

Yet another page that can be accessed through the Navigation Bar is the Connect with Others page, which will use the user’s major and try and connect them with discord servers or other group media if they are an Austin Peay student.

### Functionality

On the signup page, the site will store entered information to create a User object, storing the user’s data, as well as initialize other important features for the site, like an empty TaskList, graphs, objectives, inventory, and garden (ClassDiagram.pdf). If the user has already logged in, the site will access the server to retrieve the User object, which contains the initialized objects used for other aspects of the site.

The Dashboard page allows the user to update the TaskList through adding, removing, or editing Tasks. Removing a Task will remove the Task from the TaskList from the Dashboard and award the player. Adding or editing a Task will cause a modal to open, either with new, blank Task information or already filled in information for a preexisting Task. This includes the name of the Task, the estimated time to complete the task, rankings on difficulty and priority, and the inclusion of Subtasks. Subtasks are simpler tasks that will be included as part of Task completion. Completing a Subtask will be like completing a Task, but Subtasks will be grouped together under a Task in order.

The user is rewarded for time spent “clocked in” on a Task, completion of a Task, and completion of Daily and Weekly Objectives. The reward for time is calculated by the following:

Time XP = (minutes of work) \* (difficulty multiplier) \* (base XP constant) \* (garden multiplier)

The difficulty multiplier will be a constant value dependent on the user’s ranked difficulty of a task.

Easy = .8 Medium = 1.0 Hard = 1.2

On completion of a Task, the same amount of XP as gained during working on the Task (the time reward) will be given to the user.

On completion of the Daily or Weekly Objectives list, the reward will be the sum of the Time XP reward for each individual task added together and multiplied by .2.

The user will also be able to see two graphs related to their logged time, a Daily and Weekly graph. The Daily graph will store an array of Days, which holds arrays containing information on the data of the assignment and another which holds the amount of time worked on it in a session. The Weekly graph Object will hold an array of integers tracking the total amount of time worked on any project throughout a week. With this information, a graph will be updated on the loading of the page to display the user’s activity.

The Inventory of the User will track bought items and the state of their Garden. Users will be able to purchase items from a Shop, which will validate their purchase and update their XP and inventory. Users will be able to drag and drop items onto their Garden, which will update the information and display. Users will also be able to purchase one-time-use items which will change the Garden multiplier temporarily for a large burst of XP for their work during a set amount of time.

### Web and Server Components

## Appendix

Figure 1: Block Diagram

Figure 2: Component Diagram

Figure 3: UI Storyboard

Figure 4: Message Documentation

Figure 5: Storage Documentation