TEST PLAN + FEATURES LIST

1. File formats for user management

1.a) User data is stored in a colon-limited format and must support all Unicode languages. Each line contains data in the format "**timestamp:player:level**" where the fields are time of last play, name of the player and the last level that was attempted.

1.b) Level data is stored as "**level:sequence:rows:start:interval:decrement**". See level data table for details. "**sequence**" itself is a comma separated list (E.g. "1,1,1,2,3,1,3,1,3,3").

2. Game start up

2.a) Upon start up, the program will attempt to read a file named "**pvz_levels.csv**" in home directory. If this file does not exist or is not readable, then the program exits with an error.

□ 2.a.1 Remove "**pvz_levels.csv**" file and start the program. Observe program exiting with an error message about the level data not being available.

2.b) Upon start up, the program will attempt to read a file named "**pvz_players.csv**" in home directory. If this file does not exist or is not readable, then silently skips the read operation.

□ 2.b.1 Remove "**pvz_players.csv**" file and start the program. Observe program starting up with no users.

2.c) If the file exists, each line is read and parsed according to 1.a.

□ 2.c.1 Copy the file "**pvz_players_test-2c1.csv**" as "**pvz_players.csv**". Start the program. Observe 4 users named "Sam Iam", "□□□", "通" and "Владимир" in the user list.

2.d) If player name contain any character other than alpha-numeric characters or is longer than 10 characters, the entire file is discarded and a warning message is displayed and game starts without displaying the user name.

□ 2.d.1 Copy the file "**pvz_players_test-2d1.csv**" as "**pvz_players.csv**". Start the program. Observe a warning message about invalid player data and program starting with no users.

2.e) If the level contains any characters other than 0-9, is less than zero or is higher than 100, the entire file is discarded and proceed as in 2.b.

□ 2.e.1 Copy the file "**pvz_players_test-2e1.csv**" as "**pvz_players.csv**". Start the program. Observe a warning message about invalid player data and program starting with no users.

□ 2.e.2 Copy the file "**pvz_players_test-2e2.csv**" as "**pvz_players.csv**". Start the program. Observe a warning message about invalid player data and program starting with no users.

2.f) If the user names in 2.a was successfully read, then the last user that played (highest time stamp) is selected as current user and the last level he/she attempted is set as current level.

□ 2.f.1 Repeat test 2.c.1. Observe that the player "□□□" is selected.

□ 2.f.2 Observe current level set to 3.

2.g) If the file was not read in 2.a, current user and level is not set.

□ 2.g.1 In test 2.b.1, observe that user and level is not set.

2.h) The game displays a screen similar to figure 1 (actual look may be different, but must contain all functional parts). "User" is a drop-down list that shows last 5 users (in order of playing, from last to first). "name" is a text-area that allows a name to be typed in (initially blank). "level" is a text label that shows current level. "Delete", "New", "Start", "Restart" and "Quit" are buttons.

2.i) If the current user and level is not set, only the "New" button is active (other buttons are disabled). If the current user is set, only "Restart" and "Quit" buttons are disabled.

□ 2.i.1 In test 2.b.1, observe that the only possible action is creating a new user.

□ 2.i.2 ~~In test 2.c.1, observe that the only possible actions are "Start", "Quit", "New" or selecting another user "Restart" and "Quit" buttons~~.

2.j) When user enters a valid name (see 2.c), it will show a message box which show the name, "Ok" and "Cancel" buttons and asks user to confirm the new user. If user clicks "Cancel", a new user is not created and must keep the previous state. If user clicks "Ok", a new user is created and set as the current user. Current level is set as 1.

□ 2.j.1 Upon starting, enter a valid name. Observe a confirmation step with a choice to accept or cancel. Click cancel. Observe current user and level remain unchanged.

□ 2.j.2 Repeat test 2.j.1 and confirm creating a new user. Observe the newly created user is selected and level is set to 1.

□ 2.j.3 Upon starting, attempt to enter an invalid name. Observe that either it is not possible to enter an invalid name or that a new user is not created.

3. Game operation

3.a) Text label "points" displays the starting "sun" points and is updated whenever user collects "sun".

3.b) When User clicks "Start" button, the game starts the current level. If player survives the level, then the next level is automatically started.

□ 3.b.1 Repeat test 2.c.1. Start the game. Observe that the game starts from level 3.

□ 3.b.2 Successfully complete the level. Observe that the game starts level 4.

3.c) Once start button is clicked, all buttons except "Restart" and "Quit" buttons are disabled.

□ 3.c.1 In test 3.b.1, observe that only possible actions are "Restart" and "Quit".

3.d) After starting, if user clicks "Restart" button, game must be paused and a message box must be shown with the message "Are you sure you want to restart?" and an "Ok" and "Cancel" button. This dialogue box must cover the middle three rows and 7 columns of the 9x5 lawn grid. If user

presses “Cancel” button, game resumes from where it stopped. If User presses “Ok” button, the level is restarted.

□ 3.d.1 During game play, click “Restart” button. Observe that the game is paused and a message window with options to confirm restarting or to cancel is shown that covers the middle three rows and 7 columns of the 9x5 lawn grid.

□ 3.d.2 Select “Cancel”. Observe that the game continues from previous state.

□ 3.d.3 Repeat test 3.d.1 and confirm the restart operation. Observe that the game level is restarted.

3.e) After starting, if the user clicks “Quit” button, a dialogue box as in 3.d is shown with message “Are you sure you want to leave this level?”. “Cancel” button acts the same as 3.d. If User presses “Ok” button, the start screen (see 2.h) is shown.

□ 3.e.1 During game play, click “Quit” button. Observe that the game is paused and a message window with options to confirm exit or to cancel is shown that covers the middle three rows and 7 columns of the 9x5 lawn grid.

□ 3.e.2 Select “Cancel”. Observe that the game continues from previous state.

□ 3.e.3 Repeat test 3.d.1 and confirm the restart operation. Observe that the game play is aborted and either the program exits or goes back to start state.

3.f) “Sun” points drop at a rate of one every 10s randomly at the top of lawn columns, fall down at a speed of 1s/square and rest on a random row. After resting for 7.5s, they disappear. If user clicks on a “sun” point before it disappears, it is removed and 25 points added to the user.

□ 3.f.1 Start a game level. Observe that sun points drop at 10s intervals.

□ 3.f.2 Observe that sun points appear at random columns and drop at a rate of about 1s per square.

□ 3.f.3 Observe that sun points come to rest on a random row.

□ 3.f.4 Click on a sun point. Observe that the sun point disappears and 25 points are added to total sun points.

□ 3.f.5 Observe that sun points disappear 7.5s after coming to a rest on a given square.

3.g) Each seed box (P1-P8) must show the pictorial representation of each plant, a tool-tip message with it's name and the cost as shown in plant data table.

□ 3.g.1 Upon starting game, observe that 8 seed boxes are shown with a pictorial representation (any distinct shape).

□ 3.g.2 When mouse cursor hovers over each seed box, observe that a tool-tip message (or equivalent)
is shown that display the name and cost.

3.h) Only seed boxes for plants that are usable must be enabled and all other other seed boxes must be greyed out. Plants are not useable if the user doesn't have enough coins to purchase it or the seeding timer hasn't yet expired.

□ 3.h.1 Start a game level. Observe that all seed boxes are disabled (clicking will cause no effect) at start.

□ 3.h.2 After collecting sun points, observe that each seed box becomes active as soon as there are enough sun points to purchase a seed.

3.i) If a seed box is selectable, user may click on it and then click on a square to plant it. Seed box becomes disabled after clicking on it. The seeding timer starts as soon as the seed is planted on a square and the count down must be shown in the seed box. Plant becomes active after it's firing interval has elapsed and will fire at this rate whenever there is a zombie in its range.

□ 3.i.1 Observe that clicking on a seed box that is enabled causes to become “selected” (change appearance in some way).

□ 3.i.2 Observe that clicking on the same seed box again will have no effect.

□ 3.i.3 Click on an empty square. Observe that a representation of a plant of the selected seed is shown in the square.

□ 3.i.4 Observe that total sun points are reduced by the cost of this seed.

□ 3.i.5 Observe that the last selected seed box is shown as “unavailable” even when there are enough sun points.

□ 3.i.6 Observe that the last selected seed box stays “unavailable” for a duration as specified by the “Seeding time” of that plant, starting from planting event.

3.j) If user clicks on one seed box and then clicks on another selectable seed box, first seed box becomes enabled and the second box becomes disabled. When user clicks on a square, a plant from last clicked seed box gets planted.

□ 3.j.1 In test 3.i.1, observe that clicking on a different seed box that is enabled causes previous seed box to return to normal appearance and current seed box becomes “selected”.

□ 3.j.2 In test 3.i.3, observe that it is the last selected seed that gets planted.

3.k) Levels may have 1, 3 or 5 plantable rows, which must be centred in the lawn grid and shown in green. Non-plantable rows must be shown in brown. Plants can only be placed in plantable rows.

□ 3.k.1 Start a game and play level 1. Observe that only the middle row is shown in green. All other rows are shown in brown.

□ 3.k.2 Repeat test 3.i.3. Observe that clicking on a brown square has no effect.

□ 3.k.3 Observe that only clicking on an empty green square cause a seed to be planted.

□ 3.k.4 Start a game play with level 2. Observe that only the three middle rows are shown in green and the top row and bottom row are shown in brown.

□ 3.k.5 Repeat test 3.i.3. Observe that clicking on a brown square has no effect.

□ 3.k.6 Start a game play with level 3 or higher. Observe that all rows are shown in green.

□ 3.k.7 In tests 3.k.1, 3.k.4 and 3.k.6, observe that zombies only appear in front of green squares of the last column.

3.l) Sun producing plants produce 1 sun (25 points) at their firing rate. These stay on the same square and behaves same as in 3c.

□ 3.l.1 Start a game level. Ensure that “Sun flower” seed box is enabled and plant a sun flower on an empty square. Observe that sun points are generated on this square every 24s and they stay on this square.

□ 3.l.2 Click on a sun point generated by a “Sun flower”. Observe that the sun point disappears and that 25 points are added to the total sun points.

□ 3.l.3 Observe that a sun point generated by a “Sun flower” disappears after 7.5s.

3.m) During each level, zombies are released as indicated in the level data table and move as indicated in zombie data table.

□ 3.m.1 Start a game level. Observe that zombies are released on random rows on the last column.

□ 3.m.2 Observe that each zombie moves at a speed indicated by their “Speed” attribute.

3.n) When a zombie reaches a square with a plant, it stops and starts “eating” it at the given rate. If the plant gets destroyed (based on the health of plant and the survival of the zombie), the zombie resumes its march.

□ 3.n.1 Observe that a zombie stops when it reaches a square with a plant.

□ 3.n.2 Observe that the games shows the zombie attacking the plant in a visually identifiable way.

□ 3.n.3 Observe that each zombie attack reduces the “Life” of the plant by the amount of “Attack”.

□ 3.n.4 Observe that each zombie attacks at a rate determined by the “Rate” of each zombie.

□ 3.n.5 Observe that the plant disappears after it's “Life” becomes zero or less.

□ 3.n.6 Observe that zombie resumes its march after the plant disappears.

3.o) Plants are active after as many seconds as indicated in the “rate” column of the plant data table. They start firing as soon as zombies are within their range. Firing rate, damage, splash damage and other effects are as indicated in the plant data table.

□ 3.o.1 Start a game level and plant a “Pea Shooter” on a row where there is a “regular” zombie at least 4 squares away. Observe that the plant will not fire at the zombie for 1.5s.

□ 3.o.2 Observe that the plant fires one pea every 1.5 seconds.

□ 3.o.3 Observe that plant shots are graphically depicted in some form.

□ 3.o.4 Observe that the zombie disappears after 10 shots reach the zombie.

□ 3.o.5 Repeat test 3.o.1 with a “Snow Pea”. Observe that when the first shot reaches the zombie, it reduces the speed of the zombie by a significant amount (~50%).

□ 3.o.6 Observe that subsequent shots have no further effect on the speed of the zombie.

□ 3.o.7 Repeat tests 3.0.2, 3.0.3 and 3.0.4 with same observations.

□ 3.o.8 Repeat test 3.o.1 with a “Sun flower”. Observe that zombie reaches this square without any damage.

□ 3.o.9 Observe that the plant disappears 2s after the zombie reaches this square.

□ 3.o.10 Repeat tests 3.o.1 to 3.o.4 with all combinations of plants and zombies. Observe that they behave in line with game mechanics according to their respective attributes.

3.p) Single-use plants (“bomb” column) fire once and are then destroyed.

□ 3.p.1 Repeat test 3.o.1 with a “Cherry bomb” ensuring that there are zombies within a 3x3 region as well as outside this region. Observe that the plant “explodes” (disappears) after 1s.

□ 3.p.2 Observe that all zombies within the 3x3 square are destroyed.

□ 3.p.3 Observe that zombies outside of the 3x3 square are not damaged.

□ 3.p.4 Repeat test 3.o.1 with a “potato mine”. Observe that it does not become active (e.g. explode) for 15s after planting.

□ 3.p.5 Observe that it “explodes” when a zombie reaches the square in front and only destroys the zombie in this square.

3.q) Plant upgrades require a specific plant (“need” column) already planted in the square.

□ 3.q.1 Repeat test 3.o.1 with a “Repeater” observe that it can only be planted on a square with a “Pea shooter” in it. **[2]**

3.r) Each square in the “home column” has a single-use lawn mower. If a zombie reaches the home square, the lawn mower starts and goes through the row, destroying all zombies.

□ 3.r.1 Start a game level. Observe that all green squares of the first column contains “Lawn mowers”.

□ 3.r.2 Observe that when a zombie reaches a square with a “Lawn mower”, it destroys all the zombies in this row.

□ 3.r.3 Observe that the “lawn mower” disappears after it destroys the row of zombies.

3.s) If a zombie reaches home square that has no lawn mower, the game is lost. A message that says “Zombies have eaten your brain” is displayed and the level is restarted.

□ 3.s.1 Observe that when a zombie reaches home square that has no lawn mower, a message that says “Zombies have eaten your brain” is displayed.

□ 3.s.2 Observe that the level is restarted.

3.t) If player completes all levels, the game is won. A message to that effect is shown. Current level is set to 1 and the start screen (see 2.h) is shown.

□ 3.t.1 Complete all levels successfully. Observe that a message indicating that the game is won is shown. Current level is set to 1 and the start screen is shown.