Blackbox Testing

Requirement ID	Description of Requirement	Description of Test / JUnit Test Name	Expected Result	Result
UR_Timer	The user should be given a visual time representation relating to the singular task or order they are completing.	Tested in both scenario mode and endless mode. Timer can be seen working correctly in the top left hand corner. Also tested against stopwatch to check accuracy and is spot on.	PASS	PASS
UR_Gameplay	The user should be able to play the game as intended	Gameplay tested in scenario mode and endless mode from start to end. Works as intended.	PASS	PASS
UR_UX	The system will offer clean graphics, showing clearly what each station is, along with clearly labelled menus and buttons.	Tested through gameplay. Menu can be seen clearly showing different game mode options. Graphics are well created allowing users to understand what each station and the different ingredients are meant to be.	PASS	PASS
UR_UX	The system will offer a fully interactable and satisfactory UI for the user to control the game state, (loading into and out of the game, selecting menu buttons).	The UI is perfectly interactable however, when full screened at the start the buttons cannot be interacted with.	FAIL	FAIL
UR_Gamemodes	The user will be able to choose between two game modes; Scenario and Endless	Tested by selecting each gamemode in the menu screen. Endless mode options (easy, medium, hard) work as intended. As well as scenario mode.	PASS	PASS
UR_Customer_Counter	There will be a counter where customers will visibly give their orders and then collect them from the user within a given time frame	Tested through gameplay in both scenario and endless mode. Works as intended.	PASS	PASS

UR_Customer_Aug	The user will be able to change the number of customers that arrive (default to 5)	Tested for scenario mode. In the menu screen set the number of customers to 1,2,3 and 4. All working as planned. Game ends automatically after the last customer leaves.	PASS	PASS
UR_Loss	The user will lose the game if all reputation points are lost	Tested in endless mode by inactivity until all user reputation points are lost. Each customer that is not served causes a reputation point to be lost. Works as intended.	PASS	PASS
UR_Win	The user will win if they serve all customers without losing their reputation points	Tested in scenario mode by playing the game to completion. Players would win if all customers are served without losing all reputation points.	PASS	PASS
UR_Endless	The user will be able to choose to play an endless mode where customers will keep arriving until they lose	Option for endless mode is visible on the menu screen. Tested in all difficulty levels, through completing a few orders and then staying inactive. Customers are seen arriving after each dish is completed until the user loses from the loss of all reputation points.	PASS	PASS
FR_Power_Ups	The user will be able to acquire some five special powers	This was tested by playing the game. All 5 power-ups can be obtainable. All 5 power-ups work as intended.	PASS	PASS
UR_Save	The user can save the game	Save game feature works correctly. Tested in endless mode by completing a few orders then saving and being able to come back to where it was left off.	PASS	PASS
FR_End	The system will recognise when the	Tested through gameplay in both	PASS	PASS

	player has lost/won the game and terminated the session	scenario and endless mode. After serving all customers, the system shows the player their results and then terminates the session.		
FR_Alert	The system will notify the user when something has happened (Eg. reputation lost, chef change)	Tested through gameplay. No alert was seen when a reputation point is lost. No alert was seen for chef changes etc.	FAIL	FAIL
FR_Recipes	The system will provide the user several recipes	Recipes can be seen in the centre of the map. Users can follow the instructions for each item. Works as intended.	PASS	PASS

Whitebox Testing

UR_Chef_Swap	The user will be allowed to swap between chefs, even whilst one is completing a task.	Chef_SwapTest	PASS	PASS
UR_Chef_Move	The user can move the chefs freely around the kitchen	PlayerTest	PASS	PASS
UR_Chef_Interact	The chefs will be able to interact with any object in front of them (e.g. – Chopping board)	Chef_InteractionTest	PASS	PASS
UR_Reputation	The user will be able to lose reputation points if customers are not served	LoseRepTest	PASS	PASS
UR_Recipe	The user will make different recipes, requiring interaction with different stations	RecipeTest	PASS	PASS
UR_Investment	Users will be able to invest their earnings to unlock more chefs and stations	InvestmentTest	PASS	PASS
UR_Burning	The user will be able to burn the food if it	BurnTest	PASS	PASS

	was left to cook for too long		PASS	PASS
FR_Controls	There will be a simple control scheme in place for the user to move the chef. The control scheme will include an interact button and a button that will cycle through the controllable chefs.	PlayerTest	PASS	PASS
FR_Collision	The chefs will collide with the and not be able to move closer when in a certain range of an object	CollisionTest	PASS	PASS
FR_Interact	When the chef is within the set range of a station or object, such as the counter, they will be able to use the interact key to use said station or object.	Chef_InteractionTest	PASS	PASS
FR_Invalid_Interact	If the chef does not have the required ingredients for the station they intend to use they will not be able to interact with the station.	Chef_InteractionTest	PASS	PASS
FR_Reputation	The user will start with 3 reputation points, if they fail to complete an order in the time given they lose a point.	LoseRepTest	PASS	PASS

Non-Functional Requirement Testing

Requirement ID	Description of Requirement	Linked Requirement (therefore tested)	Expected Result	Result
NFR_reliable_gameplay	The game should rarely crash. This should have no permanent effect on the game's operation	All UR	PASS	PASS
NFR_bugfixing	Any game breaking bugs will be fixed and updated	All UR	PASS	PASS
NFR_loadtime	The game should load within a reasonable time frame	All UR	PASS	PASS
NFR_lag	The game's lag should be non-existent or at least not noticeable.	All UR	PASS	PASS
NFR_instruction_set	The game will include an easily understandable description of the aim of the game and how to operate it	UR_UX	PASS	PASS
NFR_save_corruption	If a save file becomes corrupted, the other files and the game itself will not be affected	UR_Save	PASS	PASS
NFR_control_set	The game will use simple controls which can be understood by anyone who can use a mouse and keyboard	UR_UX UR_CHEF_MOVE	PASS	PASS
NFR_tutorial	The game will	UR_UX	PASS	PASS

	contain a tutorial and a reminder screen of the controls		PASS (green)	PASS (green)
NFR_options	The hotkeys will be changeable to make the game easier to use	UR_UX	PASS	PASS
NFR_map_design	Map design will include high contrast and clearly distinguishable designs for interactable objects to make it easy to navigate	UR_UX UR_Customer_Counter	PASS	PASS
NFR_audio_visual _compatibility	Any audio cues will also have a visual cue	UR_UX	N/A	N/A
NFR_visual_focus	Pictures will be used to represent things instead of words as much as possible	UR_UX	PASS	PASS
NFR_standard_co ntrols	Hotkeys and controls will be similar to /the same as other controls for similar games to match the standard	UR_UX UR_Gameplay	PASS	PASS