

# Code Testing Plan: Growing Gamers

## Table of Contents:

1.0.....	Scope and Assumptions	
1.1 Assumptions.....		1
1.2 Environment.....		1
1.3 Within the Scope.....		1
1.4 Out of the Scope.....		1
2.0.....	Testing Scenarios	
2.1 Navigation.....		2
2.2 Data Logging.....		2
2.3 Message Generation.....		2
2.4 SMS Messaging.....		2
2.5 Accounts and Security.....		2
3.0.....	Execution Plan	
3.1 Roles and Responsibilities.....		3
3.2 Deliverables.....		3
3.3 Schedule.....		3
4.0.....	Results	

## 1.0 Scope and Assumptions

### 1.1 Assumptions

- a. The end user will be using a windows machine to run the app.
- b. The parent will most likely access the parent portal from their cell phone.
- c. Multiple users may make requests to the server simultaneously.

### 1.2 Environment

The testing environment will require:

- a. The server to be running with the terminal window in view of the tester.
- b. Overwolf to be installed and running with developer options enabled.
- c. Growing Gamers to be installed and enabled.
- d. The tester to have a cell number to receive SMS messages from the app.
- e. To have League of Legends installed and updated.
- f. To run the app on a windows 10 machine.

### 1.3 Within the Scope

Aspects of testing that are within the scope include:

- a. Navigation.
- b. Use cases.
- c. Data logging.
- d. Message generation and display.
- e. SMS messaging.
- f. Account system.
- g. Security.

### 1.4 Out of the Scope

Aspects of testing not within the scope include:

- a. Scalability.
- b. Optimization.

## 2.0 Testing Scenarios

### 2.1 Navigation

Scenarios to be tested in regards to navigation include:

- a. Navigating to the parent portal from the app.
- b. Navigating to the parent portal from the URL.

### 2.2 Data Logging

Scenarios to be tested in regards to data logging include:

- a. Logging of game data.
- b. Logging of settings data.
- c. Logging of rule violations.

### 2.3 Message Generation

Scenarios to be tested in regards to message generation include:

- a. Rule violation message generation.
- b. General message generation.
- c. Competing message generation.

### 2.4 SMS Messaging

Scenarios to be tested in regards to message generation include:

- a. Digests.
- b. Rule violations.
- c. Security codes.

### 2.5 Accounts and Security

Scenarios to be tested in regards to accounts and security include:

- a. First login on the app.
- b. Subsequent launches of the app.
- c. First login in the parent portal.
- d. Subsequent logins on the parent portal.
- e. Cell authentication for access to the parent portal.
- f. Cell authentication for login on the app.

## 3.0 Execution Plan

### 3.1 Roles and Responsibilities

Testing scenarios will be divided as follows:

- a. Shane Toma
  - i. 2.1 Navigation
  - ii. 2.2 Data Logging
  - iii. 2.3 Message Generation
- b. Bryden Trakalo
  - i. 2.4 SMS messaging
  - ii. 2.5 Accounts and Security

Each tester is responsible for:

- a. Determining useful cases and edge cases to test their assigned scenarios thoroughly.
- b. Designing and executing tests to cover the scope of the cases that were determined necessary.
- c. Reviewing and providing feedback for the cases and tests designed by the other tester prior to execution of the tests.

### 3.2 Deliverables

There are four important components to the deliverables:

- a. Cases to be tested.
- b. Tests to satisfactorily cover the cases.
- c. Results.
- d. Remediation.

Only deliverables a, b, and c should be completed in the testing phase. Deliverable d should be completed post testing. Testers should follow the sample deliverable format at the start of section 4.0 Results.

### 3.3 Schedule

It is up to the discretion of the tester to determine an appropriate schedule. The strict parameters are:

- a. The deliverable should be ready by 5:00 PM on March 29th to allow time for analysis and remediation.
- b. Testing should not take place until feature freeze which is currently scheduled to be on March 26th. Exceptions may be made if unanimously agreed upon.

## 4.0 Results

Sample Testing Scenario Deliverable			
Case	Test	Result	Remediation
Case a	Test i	Result i	Remediation i
	Test ii	Result ii	Remediation ii
	Test iii	Result iii	Remediation iii
Case b	...	...	...
	...	...	...

Navigation			
Case	Test	Result	Remediation
Navigate to the app. (Launch app automatically at startup of game)	First launch	Expected: App would launch, cell input would show Server crashed on access timestamp of games in message generation	Change the message generation to skip over game detail if no games played
	Launch with prior login	Expected: App would launch, main page displays Results were as expected	
	Second launch without prior login	Expected: App would launch, cell input would show Results were as expected	
Get back to the	Dismiss default	As expected	

game (dismiss the message).	message		
	Dismiss rule violation message	As expected	

Data Logging			
Case	Test	Result	Remediation
Log win/loss	Lose a game.	Not implemented	
	Win a game.	Not implemented	
Log kills/deaths	Get more kills than deaths	As expected	
	Get more deaths than kills	As expected	
Log rule violations	Bedtime violation	As expected	
	Playtime violation	Failed, did not log	Did an insert into the database when violated.
	Gamesplayed violation	Failed, did not log	Did an insert into the database when violated.
	All violations simultaneously	Failed, did not log playtime or games played	Did an insert into the database when violated.
Log rule changes	Parent portal rule change submission	As expected	
	Parent portal no change submission	Performed an insertion, but as expected	

Message Generation			
Case	Test	Result	Remediation
Default Welcome	At launch	As expected	

Message			
Homework Message		Not implemented	
Doing great message	Positive KD, no violation	Failed, showed take a break message.	Minor fix to the ratio() function for edge case deaths=0
Take a break message	Negative KD, no violation	As expected	
It's late message		Not implemented	
Bedtime violated message	When bedtime violated	Expected: bedtime message would be received and sent Outcome, error in message retrieval from remote. Second outcome: Bedtime only triggered after a game has been played not before Third outcome the the dismiss button message was undefined	Needed to access Backgroundcontroller's functions through the background class itself and not 'this' Second remedy: send a timestamp along with message request Third remedy: the dismissButtonMessage field had a typo, fixed that
	When bedtime and playtime violated	As expected	
Playtime violated message	When playtime is violated	As expected	Variables not set when sent
	When playtime and game limit is violated	As expected	
Game limit violated message	When game limit is violated	As expected	Variables not set when sent

SMS Messaging

Case	Test	Result	Remediation
Violation SMS	Bedtime violation	Server crashed 2nd attempt ran as expected	query needed to be set for the parent rule request in ruleSMS
	Playtime violation	As expected	
	Game limit violation	As expected	
	All simultaneous violations		
Digest SMS	Daily digest	Sent out digests constantly at 21:00	Change the trigger so its only at 21:00:00:00 that the digest is triggered (one second) not 21:00 (one hour)
	Weekly digest	Not implemented	
	Monthly digest	Not implemented	
Verification code SMS	Initial app setup	As expected	
	Parent portal login	As expected	

Account and Security			
Case	Test	Result	Remediation
Creating account	New account initial app setup	Expected: Verification code is sent to cell and access granted once code entered Output: as expected	
	Exiting account initial app setup	Expected: app display goes away As expected	



	Invalid number initial app setup	As expected	
Parent portal access	Attempt valid access with no verification code	Access denied	
	Attempt valid access with valid code that belongs to a different valid number	Access denied	
	Attempt valid access with verification code	As expected	
	Attempt access with invalid phone number	Regex doesn't work	Have the dismissal of the pop up check the regex first