Introduction  
The testing of this project was challenging. As this is a research project, the project itself is constantly evolving, with modules being added, modified and removed at an almost daily basis. Additionally, the project suffered a massive setback. Due to an accident in the RZ-11 lab, some of the installations on the development PCs were lost, which cost time to set up and reinstall. The testing plan was modified according to the changes in the project, the changing circumstances faced by the team, and the feedback received from the project advisors about the test plan. The testing could not be as thorough as we would have liked, but we still categorize it as successful as we deduced the core competencies and capabilities of our project.

Test Items:

Unit Tests:

1. Name: Inter network communication module  
   Participants: Daniyal Selani  
   Unit details: This module is responsible for sending and receiving data between modules that are located on 2 different computing nodes on the same network. The module is also responsible for encoding and decoding the messages.   
   Test details: Delay time between computing nodes is very important and the latency is to be as low as possible. We will measure round trip delay time. We define round trip delay time for compute node a to encode, send a message to compute node b, compute node b to decode the message, encode it again, send it to a, and a to decode the message. We measure this time because it is possible for our system to communicate back and forth between computing nodes multiple times to process a single frame of input. The test will be conducted 10 times and the results averaged to ensure consistency. The message sent will be a dummy array of dimension (65000, 1), and of size 65000 bytes. This message is chosen to test the module at it’s maximum payload capacity.  
   Pass/Fail Criteria: The performance of this module will be tested. We set a threshold of 10 ms max for roundtrip time, if this threshold is not matched, then this test would be categorized as a failure, and the module will be retooled/modified.   
   Results: Average round trip delay time observed: 7.4 ms < 10 ms. Test passed
2. Name: Data capture module  
   Participants: Daniyal Selani  
   Unit details: This module is responsible for capturing and transforming frames from the game GTA V to the proper dimensions that are needed as input to the different modules   
   Test details: The system works on a per frame basis. A bottleneck at the data capture module will slow the whole system down as all modules of the system depend directly or indirectly from this module. We will measure the frame rate (Frames per second) at which this module can capture and transform frames from the game. The game runs at 30 FPS. We will also observe if the module can correctly transform the frame into the correct dimensions, without introducing any noise/data corruption into the frame. The game is rendered at the resolution 1280x720 in full RGB. The module is to capture the frame at that resolution and transform it into images with the following dimensions:
   1. (160, 120, 1): This image is to be sent to the Neural Network module (Navigator). The image is grayscale, and the neural network that this input goes to, cannot handle an image of greater dimension. The neural network must work on low powered hardware, and hence cannot take in a large image.
   2. (340, 191, 1): This image is of maximum size that can be sent via a single UDP packet, while still maintaining the 4:3 aspect ratio that the various neural network models are designed to accept. This image is also grayscale.
   3. (480, 260, 3): This image is used in the data extraction module. The image is of higher resolution for more detail and is full RGB

The test will be conducted over a period of 60 seconds.  
Pass\Fail Criteria: The performance of the module and the validity of it’s output will be tested. The range for the capture rate is 25 – 35 fps. Transformed frames will be randomly sampled to test for correctness.  
Results: Frame rate observed: 29 FPS, which is within range. All sampled frames were correct. Test passed.

1. Name: Data recording script  
   Participants: Daniyal Selani  
   Unit details: This script records gameplay of GTA V for training models. It records screen capture and the corresponding action taken by user. The script records gameplay and stores it as .npy file.  
   Test details: It is important that the script records gameplay at a high enough and consistent frame rate so that the gameplay recording is smooth and without any big gaps in the recorded gameplay. The script must record the screen from the game’s resolution of 128x720 to a predetermined smaller resolution, to make the files from being too big. The size of the file is dictated by the resolution of the captured frames, and the number of datapoints in each file. As the system is expected to operate at 30 FPS, the script should ideally record gameplay at the same rate. The script will be tested by recording gameplay and storing into files with 1000, 5000, 10000 and 15000 datapoints. This is done to test the performance of the script and the ability to store data into files of different sizes. The files will then be played back to check the correctness of the data. The PC on which this script was tested was fitted with 32 GB of ram to ensure that the system does not face any bottlenecks.  
   Pass\Fail Criteria: The performance and validity of this script will be tested. The recording capture rate should be in the range of 25 – 35 FPS. The files should be stored correctly without any corruption/loss of data. The gameplay in the files will be played back to test correctness. The size of the files will also be observed  
   Results: Initially the gameplay recording FPS was as high as 45 FPS, but as the number of the datapoints collected grew, the frame rate decayed. When try to save files with 1000 and 5000 datapoints, there was no issue. The frame rate was stable and the files were stored correctly. The sizes of the files were 350 MBs and 1500 MBs on average respectively. When trying to save 10000 datapoints to a single point, the frame rate dropped substantially to as low as 8 FPS. The RAM usage of the PC was maxed out and it took several seconds for all the datapoints to be stored correctly. The resulting file was 3700 MBs on average. The PC crashed when trying to save 15000 datapoints on to a single file. All the files that were successfully stored were playable and there was no loss of data. Even though the results were less than ideal, we still categorize this test as a pass.