

Date and Time of Test Run		12/11/2017, 6:30PM					
HW Required		Arduin Mega 2560, Android Galaxy Sky, Logitech C270 Webcam, Pan Tilt Kit and Servos, UniHobby 4WD Robot Chassis					
SW Version		Python 3.6.2, OpenCV 3.0 + Contribs, Android Studio 3.0.1, Amazon Linux AMI 2017.09.1					
Test ID	Test Scenario	Platform	Test StepsTest Description (Steps)	Actual Results	Pass/Fail	Priority	Notes
AWS 1-1	Mugshots stored	Python 3.6.2, Amazon Linux AMI 2017.09.1	Store mugshots pictures on S3 bucket	Mugshots stored	Pass	Critical	
AWS 1-2	XML generated	Python 3.6.2, Amazon Linux AMI 2017.09.1	Run command to train model on stored mugshots picture and generate XML on bucket	XML generated	Pass	Critical	
ANDROID 1-1	Android: Training	Python 3.6.2, Android Galaxy Sky, Android Studio 3.0.1	Press "Training Start" button, wait for completion message	Training function successfully completed. XML file generated.	Pass	Critical	
ANDROID 1-2	Android: Tracking	Python 3.6.2, Android Galaxy Sky, Android Studio 3.0.1	Press "Tracking Start" button, wait for a moment, press "Tracking Stop"	Tracking function is running. Robot is moving.	Pass	Critical	
ANDROID 1-3	Android: Restart	Python 3.6.2, Android Galaxy Sky, Android Studio 3.0.1	Press "Tracking Start" button, wait, press "Tracking Restart", wait, press "Tracking Stop"	Tracking attempts to do face detection again.	Pass	Critical	
ANDROID 1-4	Android: Camera Reset	Python 3.6.2, Android Galaxy Sky, Android Studio 3.0.1	Press "Tracking Start" button, wait, press "Camera Reset"	Camera on robot is reset to initial position.	Pass	Critical	
ROBOT 1-1	Face tracker center moves up or down from the center	Python 3.6.2, OpenCV 3.0, Arduino Mega 2560	Move up / down wrt. the camera frame	The servo with vertical motion moves according to the movement of the face	Pass	Critical	
ROBOT 1-2	Face tracker center moves left or right from the center	Python 3.6.2, OpenCV 3.0, Arduino Mega 2560	Move left / right wrt. the camera frame	The servo with horizontal motion moves according to the movement of the face	Pass	Critical	
ROBOT 1-3	The person moves further back	Python 3.6.2, OpenCV 3.0, Arduino Mega 2560	Take a few steps behind from the camera	The cart will move forward to keep your face in the majority of the frame	Pass	Critical	
ROBOT 1-4	The person comes too near to the camera	Python 3.6.2, OpenCV 3.0, Arduino Mega 2560	Take a few steps behind from the camera	The cart will move backwards to keep your face in the majority of the frame but not let it cover the complete frame	Pass	Critical	
VIDEO 1-1	Video: Saved	Python 3.6.2, OpenCV 3.0, Android Galaxy Sky, Arduino	Go through a complete training and tracking run. View video.	Video is save and viewable. No sound.	Pass	Critical	
EDGECASE 1-1	User Obscured	Python 3.6.2, OpenCV 3.0, Android Galaxy Sky, Arduino	User obscures face for a bit while tracking	Tracking is generally able to do tracking. Face detection is unreliable.	Unreliable	Medium	Algorithm and mugshot dependent
EDGECASE 1-2	User Looks Away	Python 3.6.2, OpenCV 3.0, Android Galaxy Sky, Arduino	User looks away from camera and camera sees side view	Tracking is generally able to tracking. Face detection is unreliable.	Unreliable	Medium	Algorithm and mugshot dependent
EDGECASE 1-3	User Moves Fast	Python 3.6.2, OpenCV 3.0, Android Galaxy Sky, Arduino	User moves very fast around screen while tracking	Tracking/detection is OK but servos cannot catch up.	Fail	Medium	Servos can't move that fast
EDGECASE 1-4	User Presses Button Out of Order	Python 3.6.2, Android Galaxy Sky, Android Studio 3.0.1	User presses button in random order and not in intended sequence. (e.g. Start Tracking in middle of Training)	User is unable to press in unintended order.	Pass	High	