Test Plan

CrazyTrain

2022–12–12

Version 2.0

LINKÖPING UNIVERSITY

Status

Reviewed	Anton Bossen	2022-11-15
Approved	Filipe Barbosa	-

Project Identity

Group E-mail: antbo569@liu.se

Homepage: http://www.isy.liu.se/tsrt10/group

Orderer: Filipe Barbosa, LiU, ISY
Phone: -
E-mail: filipe.barbosa@liu.se

Customer: Linköpings universitet (LiU), ISY
Phone: -
E-mail: –

Supervisor: Daniel Arnström, LiU, ISY
Phone: -
E-mail: daniel.arnstrom@liu.se

Course Responsible: Daniel Axehill, LiU, ISY
Phone: +46 13 28 40 42
E-mail: daniel.axehill@liu.se

Participants of the group

Name	**Responsibility**	**E-mail**
Morteza Akbari	Hardware (HW)	morak752@student.liu.se
Erik Axelsson	Documentation (DOC)	eriax970@student.liu.se
Anton Bossen	Project leader (PL)	antbo569@student.liu.se
Oskar Grönlund	Simulation (SIM)	oskgr783@student.liu.se
Anton Håkansson	Design (DES)	antha842@student.liu.se
Lukas Jonsson	Software (SOF)	lukjo147@student.liu.se
Patrik Lindström	Testing (TEST)	patli821@student.liu.se
Emil Lydell	File management (FM)	emily553@student.liu.se

CONTENTS

DOCUMENT HISTORY

Version	Date	Changes made	Changes by	Reviewer
0.1	2022-09-26	First draft.	Project group	Patrik Lindström
0.2	2022-10-03	Second draft. Resolved comments from supervisor.	Project group	Patrik Lindström
1.0	2022-10-05	First version.	Project group	Anton Bossen
1.1	2022-10-06	First version. Changes due to changing requirement specification	Anton Bossen	Anton Bossen
2.0	2022-11-15	Second version.	Anton Bossen	Anton Bossen

1 INTRODUCTION

This document is used to list all the tests needed to be done in order to verify that the set requirements are fulfilled. Each test states which requirement or requirements will be tested, along with a description of how the test will be conducted.

1.1 Test Structure

In the test plan, tests of every subsystem are presented under its own section and the tests are also divided according to their priority level in the requirement specification. If a test fails, either further development is needed, or the requirement needs to be renegotiated with the customer. A procedure will be presented for each test in such a way that the test can be reproduced.

Any tests that need the drone(s) flying will first be conducted in the simulation environment. This is done to ensure that the safety of the drones. If not tested in the simulation environment first, the drones have a higher risk to be in an accident.

1.2 Definition of terms

- **Crazyflie** refer to drones Crazyflie 2.0 and 2.1, developed by Bitcraze [1]. If more than one is used, the drones will be referred to as Crazyflies.
- **Crazyradio** Communication between Crazyflies and the laptop be through Crazyradio, developed by Bitcraze [1].
- **Filter**. The filter refers to the combination of observing the states and estimating the position of the Crazyflie.
- **Graphical User Interface (GUI)** allows the user to interact with the electronics in a simplified graphical overview.
- **Inertial Measurement Unit (IMU)** is a combination of accelerometers and gyroscopes to measure acceleration, orientation and rotation speed.
- **Path** is the track laid for the Crazyflie to follow.
- **Robot Operating System (ROS)** is the operating system that is used to divide the different processes for the Crazyflies.
- **Target** means anything with a marker attached to it, Crazyflie(s) and obstacles.
- **Trajectory** is the path of the moving object(s), the object(s) being the Crazyflie(s).
- **Qualisys Camera System** is the motion capture system that enables feedback on where the Crazyflies are positioned in the room.
- **Visionen** is a large room at Linköping University where the Crazyflies are tested. The Qualisys Camera system is set up in this room.
- **Yaw angle** is the angle around the z-axis (between the x-axis and y-axis).

- **Roll angle** is the angle around the x-axis (between the y-axis and z-axis).
- **Pitch angle** is the angle around the y-axis (between the x-axis and z-axis).

2 TESTS

Below, the different tests that should be performed in order to test the requirements stated in the requirement specification [2] are presented and described.

2.1 Test of the GUI Requirements

In this section, all tests that verify if the GUI satisfies all the requirements are presented.

2.1.1 *GUI Priority 1*

In the following table, the tests for priority 1 requirements of the GUI are described.

Test	Requirement tested	Resources	Description
1	2, 3, 4, 5	Computer	Test passed if a GUI is created with an emergency stop button, a simulation start button and a simulation stop button. **Procedure:** The test will first been done by seeing if the GUI prints out to console when buttons are pressed. After integration, the GUI should command the rest of the system.
2	8	Computer	Test passed if waypoint can be set for one path. **Procedure:** Open GUI and press "Create"-tab and add a waypoint.
3	9	Computer	Test passed if waypoints can be set for multiple paths. **Procedure:** Same as test 2 but several waypoints will be added and a sequence will be created.
4	10	Computer	Test passed if it is possible to start path following for one drone. **Procedure:** Open GUI and select or create a simple mission and press start in the GUI. The test is passed if the drone starts flying in the path.
5	11	Computer	Test passed if it is possible to start path following for multiple drones. **Procedure:** Open GUI and create a multi-drone mission. Place drones in position and press start in GUI. The test is passed if all drones starts following their individual path.

2.1.2 *GUI Priority 2*

Below, the test for priority 2 of the GUI is presented.

Test	Requirement tested	Resources	Description
6	6, 7	Computer	Test passed if the GUI has one button for manual mode and one for automatic mode. **Procedure:** Let the GUI print chosen mode to the console (i.e. either a recurring print of "Manual mode" or "Auto mode"), after integration of the button's activation should switch the mode, which in turn should change the printed data to represent the selected mode.

2.1.3 *GUI Priority 3*

The following table shows the tests for priority 3 of the GUI.

Test	Requirement tested	Resources	Description
7	12	Computer	The test is passed when it is possible to select mission type "Follow the leader" and other pre-defined patterns. **Procedure:** This is done by selecting the mission type when initiating a mission. The user will be able to see goal position of each drone (e.g. in console or user interface) to verify that formation is reasonable.
8	13, 14	Computer	The test is passed when it is possible to save and load a mission. **Procedure:** This is done by creating a mission in GUI with at least a starting point, a goal point and mission name. By pressing the "Save mission" button before starting the mission, the program will save the mission with given mission name in the mission-folder of the project. Likewise in GUI, there will be a "Load mission" button which will allow the user to enter a mission name. By entering a mission name corresponding to an existing mission file will load the mission. By then starting the mission will we be able to verify if it loaded properly.

2.2 Test of the Simulation Environment Requirements

In this section, all the tests that make sure that the Simulation Environment satisfies all the requirements are presented.

2.2.1 *Simulation Environment Priority 1*

In this table, the tests for the Simulation Environment of highest priority will be described.

Table 1: Simulation Environment priority 1

Test	Requirement tested	Resources	Description
9	15	Computer	Test passed when a planned trajectory is visualised matching the path requested in the GUI. **Procedure:** Open the GUI and select a planned path. When the planned path is selected, a trajectory can then be visualised for a single drone in a 3d-plot and verified with the trajectory matrix. Alternatively, plotting each axis vs time in three separate plots.
10	16	Computer	Test passed when a planned trajectory for two or more drones is visualised matching the paths requested in the GUI. **Procedure:** Open GUI and select multi planned paths. Plot the planned trajectory for multiple drone in a 3d-plot and verify it with the trajectory matrix. Alternatively, plotting each axis vs time in three separate plots for each drone.
11	17	Computer, Crazyflie, Visionen	Test passed when an estimation of the drone position is visualised in the simulation environment. **Procedure:** Open GUI and start the simulation environment by pressing "Start Simulation". Apply a simple control command such that the drone is expected to move (e.g., moving forward or upwards). Verify that the drone is moving in the expected direction in the simulation environment.
12	18	Computer, Crazyflie, Visionen	Test passed when an estimation of the drones' position is visualised in the simulation environment. **Procedure:** Start the simulation environment. Apply a simple control sequence (hovering, for example) for the fleet of actual drones. Verify that all the active drones show up in the simulated environment and that they are moving in the expected direction.

Continued on next page

Table 1: Simulation Environment priority 1 (Continued)

13	20	Computer, Crazyflie, Visionen	Test passed when the simulation is being updated in real-time. This is done by printing data in the console or plotting a figure for each cycle in the program. **Procedure:** Start the simulation environment. Apply control commands and/or a control sequence to the Crazyflie drone(s). Compare updates in the simulation to real time data printed in the console of the CPU to determine test passing.
14	21	Computer	Test passed when an emergency stop is simulated in the simulation environment. **Procedure:** Start the simulation environment. Apply a control sequence to either the Crazyflie drone(s) or to the simulated drone. Activate the emergency stop button with the GUI during flight. This should modify the planned path of the drone(s) to a landing manoeuvre. Whether the landing manoeuvre is shown in the simulated environment or not determines the test passing.
15	27	Computer	Test passed when the simulation is accessible from the GUI. **Procedure:** Start the GUI, thereafter press the button dedicated to opening the simulation environment. Study if simulation environment is opened and started.

2.2.2 *Simulation Environment Priority 2*

In the following table, the simulation tests of priority 2 are presented.

Test	**Requirement tested**	**Resources**	**Description**
16	28	Computer, Visionen, Crazyflie	Test passed when a margin of the drone position error is calculated and displayed. **Procedure:** Start the simulation environment and look at one drone. The actual position minus the position in the simulation environment shoudl be displayed as the position error.

2.2.3 *Simulation Environment Priority 3*

In the following table, the tests of priority 3 for the simulation are presented.

Test	Requirement tested	Resources	Description
17	19	Computer, objective person	Test passed if a person outside the project group can identify the drone in the simulation as a drone by its visualisation. **Procedure:** This will be tested by asking the project's supervisor if the drone in the simulation looks like an actual drone.
18	22	Computer	Test passed if the simulation results can be saved. **Procedure:** Start the simulation environment. Apply a control sequence that can be carried out in the simulation. Use the save function in the simulation environment and confirm that a csv-file with the simulated trajectory has been saved.
19	23	Computer	Test passed if a saved simulation can be loaded and played again. **Procedure:** Start the simulation environment. Selected a csv-file including a previously simulated trajectory. Check if the simulation is replayed within the environment.
20	24	Computer	Test passed if the simulation model displays attitude properties like yaw-, pitch- and roll-rate. **Procedure:** Start the simulation environment. Apply a simple control sequence that includes rotational turn in the roll, pitch and yaw dimension. Observe the output of the simulation environment, presented data for yaw-, pitch- and roll-rate concludes test passing.
21	25	Computer, Crazyflies, Visionen	Test passed if the simulation model is within one meter of the drone in real-time. **Procedure:** Start the test in a known location, then fly the Crazyflie a given path and then land the Crazyflie. Compare the simulated result with both the flown path and the set landing spot. If it deviates less than 1 meter, the test is passed.
22	26	Computer	Test passed if it can be confirmed in the simulation environment that no collision occurs. This test can only be carried out when previous tests confirming the accuracy of the simulation have been passed. **Procedure:** Start the simulation environment. Initiate drone missions that will cause the drones to collide. Observe if the drones keeps a safety distance to each other during the simulation of the colliding missions. Finishing of the missions without collision and/or "near"-misses concludes test passing.

2.3 Test of the Route Planner Requirements

All tests regarding the route planner are presented below.

2.3.1 *Route Planner Priority 1*

In this section, the test for the highest priority requirement for the planner is presented.

Test	Requirement tested	Resources	Description
23	29	Computer	The test is passed when the planner can create a trajectory from starting point to goal point with waypoints from the GUI acting as inputs and sending trajectory as output to the controller. **Procedure:** Start the GUI and set a path. Start the simulation environment and make sure it creates a trajectory. By printing input data and output data from planner (e.g. in console) we can verify if it behaves properly. After integration, the signal should go to the planner.

2.3.2 *Route Planner Priority 3*

The tests regarding the lower priority requirements for the route planner are stated in this section.

Test	Requirement tested	Resources	Description
24	30	Computer	The test is passed when it is possible to create more complicated flight programs in the planner. Examples of more complicated flight programs are "Follow the leader" and being able to create patterns with several drones. **Procedure:** Same as test 23, but create more advanced paths and for multiple drones.
25	31	Computer, Crazyflie, Visionen	The test is passed when the route planner give out a trajectory that navigates around an obstacle noticed by the sensor system. **Procedure:** By placing an obstacle in front of the drone in the simulation environment, we will see the planner will calculate a trajectory around the obstacle. When the test is performed successfully in the simulation environment, the test will be carried out in Visionen. This is done by defining a virtual obstacle in Visionen and insert waypoints that would make the drone fly through the obstacle. The planner should be able to calculate a trajectory around which the drone will follow. *

* means that the test has been changed after test plan has first been approved.

2.4 Test of the Controller Requirements

In this section, the requirements for the controller are going to be tested.

2.4.1 *Controller Priority 1*

The list of the requirements of the controller with priority 1.

Table 2: Controller priority 1

Test	Requirement tested	Resources	Description
26	32	Computer, Crazyflie, Gampepad, Visionen	The test is passed when one Crazyflie can fly manually with gamepad. **Procedure:** Start up the Crazyflie, place it in Visionen. Select manual flight mode from the GUI. Select the preferred gamepad and conclude if the drone is able to fly according to gamepad inputs. Desired behaviour of the drone concludes test passing.
27	33	Computer, Crazyflie, Visionen	The test is passed when one Crazyflie can fly autonomously. **Procedure:** Start up the Crazyflie and the supporting system (Qualisys, ROS etc.). Enter a desired path for the drone, either to the GUI directly or via a separate control sequence. Apply the control signals to the drone and observe if the decided mission can be carried out via autonomous flying.
28	34	Computer, Crazyflies, Visionen	The test is passed when multiple Crazyflies can fly autonomously and synchronised. **Procedure:** The same test as test 27, but with multiple drones.
29	35	Computer	The test is passed if it is possible to switch between manual and autonomously flying for one Crazyflie in the GUI. **Procedure:** By pressing the button for manual mode, the GUI will switch output from autonomous mode to manual mode. By pressing the button for autonomous mode, the GUI will switch output from manual mode to autonomous mode. If the drone is already in requested mode, nothing will happen. We can verify functionality by displaying outputs in console.

Continued on next page

Table 2: Controller priority 1 (Continued)

Test	Requirement tested	Resources	Description
30	36, 37, 38	Computer, Crazyflie, Visionen	Technically six different tests (one for each direction). The test is passed when one Crazyflie can move in its own local coordination system (xyz-axis) independently of its position in the global coordinate system. **Procedure:** By pressing the button assigned to moving right or left, the drone should move right or left along its local x-axis. By pressing the button assigned for forward and backward, the drone should move forward and backwards (along its local y-axis). By pressing the button assigned for up and down, the drone should move up and down (along its local z-axis).
31	42	Computer, Crazyflie, Visionen	The test is passed when one Crazyflie can rotate relative to its z-axis. **Procedure:** This will be done by rotating the yaw angle by pressing the rotate button while the drone is hovering in a stand-still.
32	44	Compute, Crazyflie, Visionen	The test is passed when a Crazyflie can follow a given trajectory with a maximal deviation of 10 cm without respect to the yaw, pitch and roll of the drone. **Procedure:** This is done in Visionen with a visualised estimated position. A simple mission of following a straight path is created. If the estimated position show that the drone doesn't drift further than 10 cm from its reference path, the test is passed.
33	45	Computer, Crazyflie, Visionen	The test is passed when the Crazyflie can hover in a specified position with a maximum deviation of 10 cm. **Procedure:** A simple mission is created where the drone should move straight up one meter and then hoover. If the estimated position show that the drone doesn't drift further than 10 cm from its reference point, the test is passed.
34	46	Computer, Crazyflies, Visionen	The test is passed when multiple drones can fly synchronised while following a path. **Procedure:** Start up the Crazyflies and the supporting system (Qualisys, ROS etc.). Apply a desired formation control sequence. Observe if the decided mission is carried out via autonomous flying. Also compare position estimate to the mission trajectories to conclude test passing.

2.4.2 *Controller Priority 2*

The test regarding the controller for the mid-priority test is presented below.

Test	Requirement tested	Resources	Description
35	43	Computer	The test is passed when the parameters for the controller can be changed. **Procedure:** This will be tested by changing the parameters of the controller and simulating the flight inside the simulation environment.

2.4.3 *Controller Priority 3*

The test regarding the priority 3 requirements is presented below.

Test	Requirement tested	Resources	Description
36	39, 40, 41	Computer, Crazyflie, Visionen	Technically six different tests (one for each direction). The test is passed when the Crazyflie is able to move respectively in the global x-axis, y-axis and z-axis independent of the yaw angle. **Procedure:** Use the GUI to create missions with two waypoints in straight paths in each direction. The test is passed if the drone is able to follow each axis independently.

2.5 Test of the Sensor System Requirements

Below, all tests regarding the sensor system are presented.

2.5.1 *Sensor System Priority 1*

In this section, the tests for the highest priority requirements for the sensor system are presented.

Table 3: Sensor System Priority 1

Test	Requirement tested	Resources	Description
37	48	Computer, Crazyflies, Visionen	Test passed when the Crazyflie is moved from one place to another and the sensor system gives a new position that is within 1 cm of the real drone. **Procedure:** This is done by moving the drone manually a small distance (10 cm), which is measured physically. The measurement is then compared to the Qualisys position with the filtered signal.
38	49	Computer, Crazyflies, Visionen	Test passed when a Crazyflie moves along a straight line and the estimated position follows the line within 4 cm from the Qualisys position. **Procedure:** The test will be done by placing a drone on a movable table. The table is then moved a physically measured distance of 2 meters. Each 50 cm that is travelled, the position is compared with the filtered position estimate. The test is passed if the position estimate error is less than 4 cm at all measurement points.
39	50	Computer, Crazyflies, Visionen	Test passed when the sensor system gives a warning when Qualisys system warns that the position estimation is too bad. **Procedure:** Start Qualisys and hold a drone inside Visionen and move the drone to a badly calibrated region. The sensor system should send out a warning if we are using a filter with bad positions outside of target precision.
40	51	Computer, Crazyflies, Visionen	Test passed if the different drones show up in Qualisys with corresponding names. **Procedure:** Test is made with multiple drones in Visionen. When a drone's active markers are covered, the names of the drones in the Qualisys system should not be changed.
41	52	Computer, Crazyflies	Test passed if the IMU data is used in the filter. **Procedure:** Verifying that the IMU data is used in the code for the filter.

Continued on next page

Table 3: Sensor System Priority 1 (Continued)

42	53	Computer, Crazyflies	Test passed if there exists a filter that combines data from Qualisys and the IMU. **Procedure:** Verifying that the IMU data is used in combination with the data from Qualisys in the code for the filter.
43	54	Computer, Crazyflies, Visionen	Test passed if the Crazyflie is rotated in the different ways and the positioning data is within 15 degrees. **Procedure:** For the yaw angle, the filtered signal will be compared with a visualised approximation of the true yaw angle. For roll and pitch angle, the filtered signal will be compared with the position in Qualisys.

2.6 Test of the Performance Requirements

All the tests regarding the performance requirements are presented below.

2.6.1 *Performance Priority 2*

In this section, the tests for the mid-priority requirements are presented.

Test	Requirement tested	Resources	Description
44	55	Computer, Crazyflie, Visionen	The test is passed when the Crazyflie can go from one point to another, with a distance of 1 m within 2 seconds. **Procedure:** This will be tested by inserting two different waypoints (1 meter apart) into the GUI. The height (z-value) will be constant for the test. The time for the movement will be calculated from the plot that the computer gives out.
45	56	Computer, Crazyflie, Visionen	The test is passed when the Crazyflie settles to stable hoovering within 2 seconds after traveling 1 m. **Procedure:** Same procedure as in test 44.
46	57	Computer, Crazyflies, Visionen	The test is passed if the Crazyflie is stable while wind is affecting the drone.* **Procedure:** Test will be done by letting a drone hover and introduce wind by faning with a book or a similar object.*

* means that the test has been changed after test plan has first been approved.

2.6.2 *Performance Priority 3*

In this section, the lower priority requirements of the performance are presented.

Test	Requirement tested	Resources	Description
47	58	Computer, Crazyflies, Static object, Markers, Visionen	The test is passed when the Crazyflie is able to avoid collisions with other non-drone targets if the targets are wearing the appropriate sensors. **Procedure:** This will be tested by adding a virtual obstacle to Visionen. Afterward, a mission will be created for a drone so that the obstacle is in the way of the planned path. The planner should then create a trajectory that avoids the obstacle.*

* means that the test has been changed after test plan has first been approved.

2.7 Test of the Modularity Requirements

All tests regarding modularity are presented below.

2.7.1 *Modularity Priority 2*

In this section, the mid-priority requirements for modularity are presented.

Test	Requirement tested	Resources	Description
48	59	Computer	The test is passed when at least two different controllers can control the system when implemented independently, and it is possible to switch between these controllers. **Procedure:** In settings, the user should be able to switch between which controller is going to be used (e.g. in a settings file before running the main program). All controllers should work properly without the user having to do anything else beside choosing which one to use.
49	61	Computer	The test is passed when at least two different planners can be used in the system when implemented independently and it is possible to switch between these planners. **Procedure:** Same as test 48, but with planners instead of controllers.

2.7.2 *Modularity Priority 3*

In this section, the lower priority requirement for modularity is presented.

Test	Requirement tested	Resources	Description
50	60	Computer	The test is passed when at least two different filters can be applied to the system when implemented independently, and it is possible to switch between these filters. **Procedure:** Same as test 48, but with filters instead of controllers.

2.8 Test of Safety and Security Requirements

The tests regarding the safety and security requirements are presented below.

2.8.1 *Safety and Security Priority 1*

In this section, the test for the highest priority requirement for safety and security is presented.

Test	Requirement tested	Resources	Description
51	63	Computer, Crazyflie, Visionen	The test is passed when a Crazyflie can land safely when the manual stop button is pressed. Safely implies a smooth landing that does not damage the Crazyflie or its surroundings. **Procedure:** The stop button in the GUI will be pressed while the drone is flying.

2.8.2 *Safety and Security Priority 2*

In this section, the mid-priority requirements for safety and security are presented.

Test	Requirement tested	Resources	Description
52	65	Computer, Crazyflie, Visionen	The test is passed if a Crazyflie can land safely when there is an error of more than 1 meter between the planned trajectory and the actual position. **Procedure:** The test will be carried out by modifying the controller temporarily and thereby creating a constant error for the controller. A simple mission to follow a straight line is created, and the error in the controller should make the drone deviate from the path. When the position error surpasses the threshold, the emergency script should be run, which then tells the Crazyflie to land safely.
53	70	Computer, Crazyflies, Visionen	The test is passed when requirement 65 can be carried out for multiple Crazyflies. **Procedure:** The test will be done in the same way as test 52, but with multiple drones. Two drones are given missions to follow parallel paths, and if one drone deviates more than the threshold from its path, both drones should land.
54	66	Computer, Crazyflie, Visionen, Cardboard	The test is passed when the Crazyflie is able to safely land when it is outside the Qualisys positioning range system. **Procedure:** An area in Visionen will be blocked with a large cardboard box, making this area outside the Qualisys cameras' reach. The Crazyflie will then fly into this area and- if the test is successful-land safely.
55	71	Computer, Crazyflies, Visionen, Cardboard	The test is passed when requirement 66 can be carried out for multiple Crazyflies. **Procedure:** The test will be carried out in two steps. Firstly, a larger area will be blocked out for the Crazyflies, which will all perform the same operation as for the case with one Crazyflie. Secondly, several areas will be blocked out in Visionen and the Crazyflies will individually fly into a blocked area and then, when out of the Qualisys cameras' reach, they will land safely.

2.8.3 *Safety and Security Priority 3*

In this section, the lower priority requirements for safety and security are presented.

Test	Requirement tested	Resources	Description
56	64	Visionen, Computer, Crazyflie	The test is passed when a Crazyflie can land safely when the connection to the CPU is lost. **Procedure:** The radio will be disconnected during flight and the Crazyflie shall be able to land safely.
57	67	Visionen, Computer, Crazyflie	The test is passed when a Crazyflie can avoid colliding with each other. **Procedure:** This is tested by inserting waypoints that would create intersecting trajectories for two drones. Check planner for the drones if the planned
58	68	Visionen, Computer, Crazyflie	The test is passed when a Crazyflie can autonomously land safely when the battery is getting too low to fly. **Procedure:** Test will be done by letting a drone with low battery fly autonomously and see if it will land.
59	69	Visionen, Computer, Crazyflies	The test is passed when multiple Crazyflies can land safely when the connection to the CPU is lost. **Procedure:** The radio will be disconnected during flight, as in the case of one Crazyflie.