Contents

1 Problem description

The problem description is taken from the project offer on blackboard[1]

The market for unmanned surface vessels (sailing drones) is growing, and there is a need for an open architecture auto-pilot that is usable on different platforms.

This project is about designing and implementing the electronics and embedded software for an autopilot which is able to receive steering commands from a navigation system (distance to go left/right) and from thoes instructions, steer a small boat (et. control left/right thrusters).

Project details:

- PCB design of autopilot electronics (if something existing cannot be used)
- Implement PID control loop to get back on course based on received left/right commands
- Control of thrusters in case of two-thruster catamaran
- Control of wheel in case of outboard motor on boat
- Test trials at sea
- EIVA will provide necessary hardware, include test boats / catamarans, including PCB production cost
- EIVA will provide work-space (desk, lunch etc) at EIVA's facilities
- IF the project is successful, EIVA will offer to finalise the product and make it available for sale with a commision to the student

2 Requirements specification draft

The requirements for the product are prioritized using the MoSCoW method. Using this, the requirements for the product are divided into four sections, where the most important elements are given the highest priority. **Must** are the requirements that are an absolute necessity for the product. **Should** are the requirements that are also of high priority, but not quite mandatory. **Could** are requirements which may be met, if the time and other constraints of the project allow for it. **Won't** are the requirements the product will not meet, but could be developed at a later point in its lifetime.

The following priorities have been chosen for this project:

Must
- Navigate waypoints from user input
- Be compatible with NMEA protocol GPS input
- Use GPS for localization
- Implement a PID control loop

Should
- Control thrusters in two-thruster catamaran
- Use a graphical user interface for user interaction
- Be able to change the PID parameters

Could
- Control wheel in outboard motor on boat
- Be generic enough to use with other engine types

Won't
- Use pylogon-coverage for a specified area
- Avoid obstacles

The following two diagrams specify the actor context to the system, and the use cases for the system.

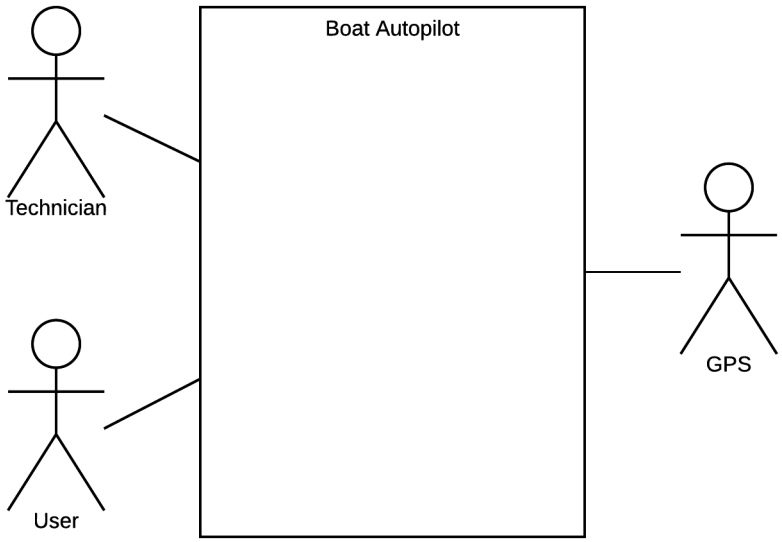

Figure 1: Actor context diagram

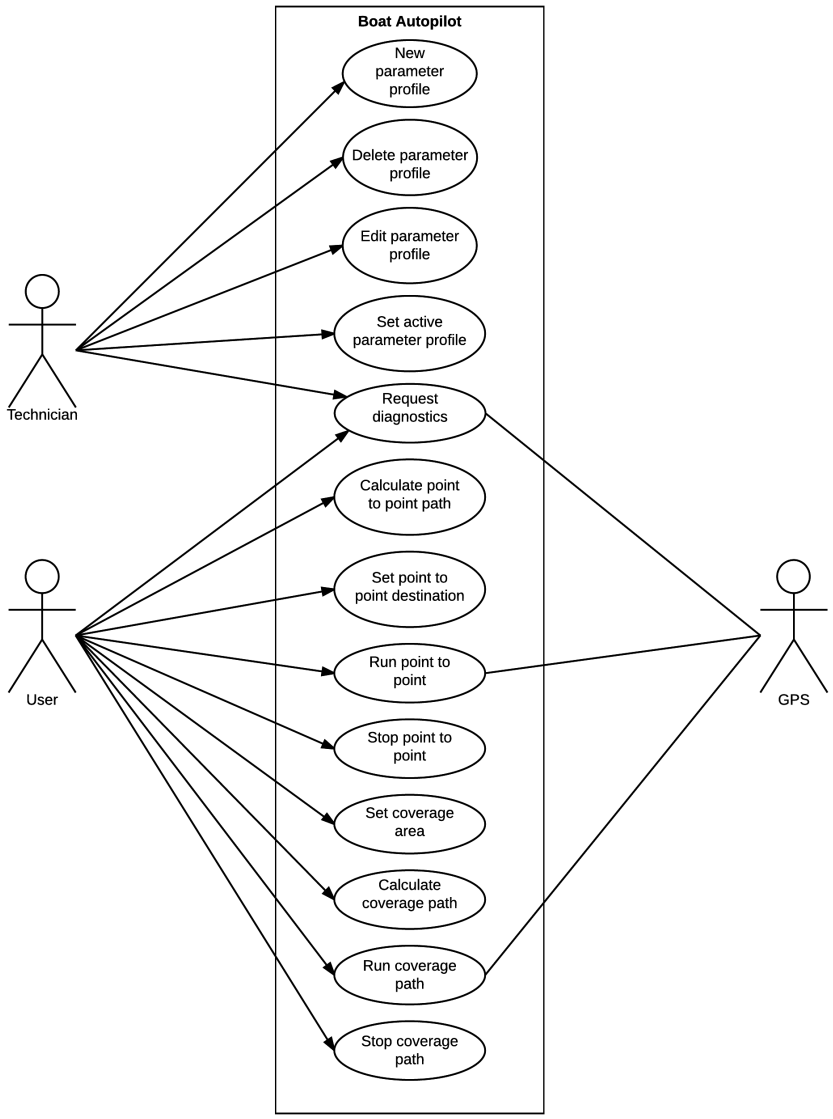

Figure 2: Use case diagram

2.1 Actor description

The following section describes the system's actors. Every actor has a type and a short description of its function and impact on the system.

2.1.1 Actor: Technician

Type:

Primary

Description:

A technician with knowledge of the system. This actor triggers the use case "Specify parameters".

2.1.2 Actor: User

Type:

Primary

Description:

The user, or customer. This actor triggers the use case "Specify waypoints".

2.1.3 Actor: GPS

Type:

Secondary

Description:

The global positioning system. This actor assists in carrying out the use case "Navigate waypoints".

2.2 Fully dressed Use cases

2.2.1 Use case 1 - New parameter profile

Goal:

To create a new parameter profile for the controller

Initialization:

Technician

Actors:

Technician (primary)

References:

Use case 3 - Edit parameter profile

Simultaneous occurrences:

One

Prerequisite:

Edit parameter menu is opened in user interface

Result:

A new parameter profile has been created.

Main scenario:

1. The technician selects "New Profile" button.
2. A new parameter profile is added to the list of parameter profiles, all parameters are set to default values.
3. The new parameter profile is passed to use case 3 - Edit parameter profile

2.2.2 Use case 2 - Delete parameter profile

Goal:

To delete a parameter profile for the controller

Initialization:

Technician

Actors:

Technician (primary)

References:

None

Simultaneous occurrences:

One

Prerequisite:

- Edit parameter menu is opened in user interface
- A parameter profile must exist.

Result:

A parameter profile has been deleted

Main scenario:

1. The technician selects a desired parameter profile to delete
2. The technician selects "Delete Profile" button.
3. The selected profile is deleted from the list of parameter profiles.

Exception:

3. a) If the active profile is deleted, no active profile is set.

2.2.3 Use case 3 - Edit parameter profile

Goal:

To edit an existing parameter profile

Initialization:

Technician

Actors:

Technician (primary)

References:

None

Simultaneous occurrences:

One

Prerequisite:

- Edit parameter menu is opened in user interface
- A parameter profile must exist.

Result:

The parameter profile has been edited

Main scenario:

1. The technician selects a desired parameter profile to edit
2. Then the technician selects "Edit Profile"
3. The selected parameter profile values are displayed on the user interface
4. The technician edits the parameter values of the profile
5. The technician selects "Save"
6. The old parameter profile is overwritten with the new values.

Extension:

5. a) The technician selects "Revert"
   b) The old parameter profile is not overwritten. The old values are kept
   c) The user interface displays the old values

2.2.4 Use case 4 - Set active parameter profile

Goal:

To set a parameter profile as active.

Initialization:

Technician

Actors:

Technician (primary)

References:

None

Simultaneous occurrences:

One

Prerequisite:

- Edit parameter menu is opened in user interface
- A parameter profile must exist.

Result:

The parameter profile has been set as active

Main scenario:

1. The technician selects a desired parameter profile to activate
2. The technician then selects "Set Profile"
3. The selected parameter profile has been set as active.
4. The name of the active profile is displayed on the user interface.
5. The boat parameters are updated with the newly activated parameter profile values.

2.2.5 Use case 5 - Request diagnostics

Goal:

To display all diagnostics data

Initialization:

Technician or user

Actors:

- Technician (primary)
- User (primary)
- GPS (secondary)

References:

None

Simultaneous occurrences:

One

Prerequisite:

None

Result:

All diagnostics data are displayed on the user interface

Main scenario:

1. The primary actor selects the "Diagnostics" menu in the user interface.
2. The diagnostics data is displayed on the user interface.

2.2.6 Use case 6 - Set point to point destination

Goal:

To set a point to point destination

Initialization:

User

Actors:

User (primary)

References:

None

Simultaneous occurrences:

One

Prerequisite:

- The user interface has an updated map
- The point to point menu is opened in user interface

Result:

A destination has been defined.

Main scenario:

1. The user selects a desired position on the map.
2. The user interface displays the selected position on the map
3. the user interface displays the selected position as coordinates

Alternate flow:

1. a) The user inputs latitude and longitude in the user interface
   b) The main scenario is continued from 2

Bibliography

[1] EIVA. *ELECTRONICS/SOFTWARE - Boat autopilot*. 2017.