

McMASTER UNIVERSITY

HYDROSWARM

SOFTWARE & MECHATRONICS CAPSTONE

Development Process & Implementation

Authors:

Victor Velechovsky - 001305263

Gabriel Potter - 001429884

Amandeep Panesar - 001431191

Taha Mian - 001417172

Nishanth Balamohan - 001411319

Professor:

Dr. Alan Wassyng



Last compiled on November 29, 2018

Contents

1	Introduction	2
1.1	Project Overview	2
1.2	Naming Conventions and Terminology	2
1.3	Roles & Responsibilities	3
1.4	Meeting Schedule	3
1.5	Communication Pipelines	4
1.6	Handling Change	4
2	Version Control	4
2.1	Technology	4
2.2	Workflow	4
2.3	Setup	5
2.4	Versioning Scheme	6
3	Process Workflow	6

List of Figures

1	Revision History	1
2	Git Workflow: Two stage release with feature-branch	5
3	6

Date	Revision	Comments	Author(s)
Nov. 29	1.0	Template added	Amandeep Panesar
Nov. 29	1.1	Added sections: Introduction, Team Members, Version Control, Workflow	All authors
Nov. 30	1.2	Final revision	All authors

Figure 1: Revision History

1 Introduction

1.1 Project Overview

HydroSwarm is a swarm of autonomous robotic boats meant to carry out measurements over large bodies of water. For our project, these boats will be measuring water temperature, but the idea can be expanded to any number of other quantifiable measurements. Central to our work will be two major components. First, a small motorized boat, attached with a water temperature sensor, as well as a control unit that allows it to communicate with and be controlled by a centralized control unit. Second, a software package that can control a large group (swarm) of these boats, with an algorithm focused on producing reliable, accurate, and fast measurements.

Our swarm will aim to cover large areas more quickly and cost-effectively than traditional products. To test the applicability of our project, we will demo it on a small scale body of water, such as a swimming pool, as well as develop a simulation to hypothetically prove the efficacy of the system on a larger scale.

Our project will be conducted between Fall 2018 - Winter 2019 for our Engineering Capstone project at McMaster University, under the guidance of Dr. Alan Wassing. We have four Software Engineering students, and one Mechatronics Engineering student.

1.2 Naming Conventions and Terminology

- **System:** The entire software and hardware package - including the boats, boat hardware, control software, and server running the control software
- **Swarm:** A large group of objects (in our case, motorized boats) that can communicate and perform acts as a group
- **Insect:** A member of the swarm (in our case, a single motorized boat)
- **Simulation:** The simulation will be used for demo purposes, mainly to show that the system is valid with a large number of insects.
- **Researcher:** A user that is interested in the data that is returned from the survey.
- **System Administrator:** A user that controls the parameters of the **swarm**.
- **G.P.S.:** Global Positioning System

1.3 Roles & Responsibilities

See Table 1 for a breakdown of the roles and responsibilities that have been established thus far in the project. As more technical and organizational challenges arise, they will be relegated to the team member(s) best suited to tackle the given task.

Table 1: Roles & Responsibilities Breakdown

Role	Member(s)	Responsibility
Algorithm Development	Victor Velechovsky	Development of path planning algorithms Evaluation of algorithms for swarm control purposes
Hardware Communication	Amandeep Panesar Taha Mian	Developing methods for swarm communication Evaluation of methods for both unit to unit communication and communication with master control
Hardware Control	Gabriel Potter Nishanth Balamohan	Develop hardware control systems for swarm units Evaluate effectiveness of control mechanisms for speed, steering, etc.

1.4 Meeting Schedule

Our team currently has weekly meetings on Thursday at 11:30 for approximately 1 hour and Friday at 11:30 for 3 hours. All group members are expected to attend them. These meetings are used as progress updates for the group. During each meeting, all group members reflect on the work that they have completed in the past week and share any updates. Following the updates, the group discusses the progress and any deadlines that require attention. The rest of the time is used for planning for the following week's tasks. The group also meets for unscheduled meetings if necessary. When there are deadlines approaching and there isn't enough progress on the tasks, the group may meet up to complete them. The team also meets with Dr.Wassyng and the TAs on a weekly basis. These meetings are scheduled for a random time each week and are used as progress updates and for clarifications.

1.5 Communication Pipelines

Communication within the team will be done primarily on Facebook Messenger and Slack. Meeting organization will be done through Facebook for usability and to ensure fast response times from team members. More detailed, in depth technical conversations will take place on Slack.

Communication with either Dr. Wassyng or the teaching assistants shall be done via email and in person. When done over email, it will be protocol to CC the rest of the team to ensure visibility on the conversation. For in-person meetings, information will be relayed to the absent team members over Slack.

1.6 Handling Change

If any part of the project has to be changed there will be a group meeting pertaining to the change. All group members are involved in the decision-making process and each member of the group has the opportunity to express their thoughts on the change and propose possible solutions. Next, a vote is held on the different solutions to see which one the group wishes to move forward with. The team will vote on the proposed changes and will act accordingly. If the group has difficulties voting on certain aspects, a teaching assistant or Dr. Wassyng will be involved to aid in the decision. Once the change has been accepted the group will work to make sure all changes are completed and reflected appropriately in the documentation.

2 Version Control

2.1 Technology

We will be using *git* version control with a front-end interface on McMaster's self hosted *gitlab* instance (gitlab.cas.mcmaster.ca). *Gitlab* will be the main source for code storage, issue tracking, and build releases. We will use the issue tracking feature to publish new tasks that need to be completed, discuss them, and assign them to group members.

For other communication purposes, we will use a combination of *Slack* and Facebook Messenger.

2.2 Workflow

The workflow our team will be using follows the feature-branch workflow. There will also be a two-stage release process involving the master branch which will receive changes from the development branch (see Figure 2). All features will be completed in their associated branches which are branched from the development branch. A pull request can be made from the feature branch into the development branch to add changes, once the code has been

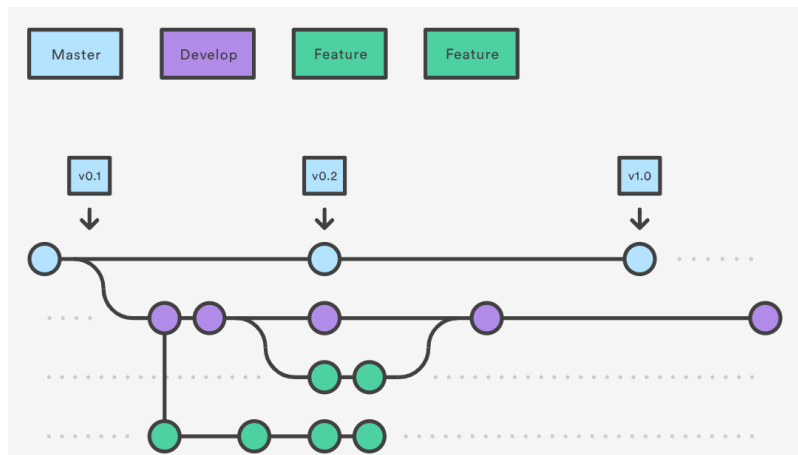


Figure 2: Git Workflow: Two stage release with feature-branch

reviewed by a maintainer (group member). Once all the necessary changes are complete on the development branch and the code is ready for production the changes will be added via a pull request into the master branch. The master branch is the branch that will be checked out for any project deliverables or demonstrations. Any code built from the master branch will be assigned a version number in the "major.minor.patch" format. A corresponding version tag will be generated at that commit hash so that the build can be easily duplicated later on.

2.3 Setup

The steps below outline how to contribute source code to the project repository

- Clone the Project

```
git clone https://gitlab.cas.mcmaster.ca/velechva/capstone.git
```

- Add a Feature Branch

```
git pull
git checkout develop
git checkout -b BRANCH_NAME
```

- Push Changes to Repository

```
git add FILE_NAME
git commit -m "MESSAGE"
git push
```

Rapid Application Development (RAD)

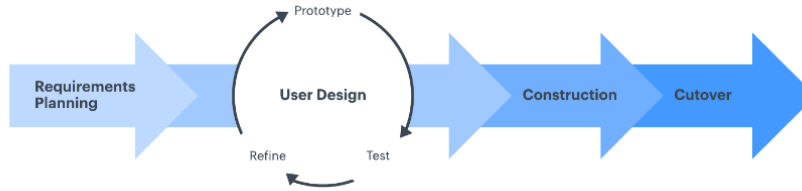


Figure 3: Rapid Prototyping Development Processes

2.4 Versioning Scheme

The version number for the system will be updated every time a feature is revised or added. The scheme for versioning will follow the *major.minor.patch* convention. The major number will increment by one when a new feature is introduced, or specifications for an existing feature are changed. The minor number will also increment by one, however only when features are refined. Fixing bugs will change the patch number, incrementing it by one.

3 Process Workflow

The development of systems will be spread out over the upcoming two months with three hour sessions every Friday. Work that is not completed will either cause the current session time to be extended, or be pushed to the next session. The team will also meet regularly on Thursdays to plan any work required for upcoming deadlines, and schedule them accordingly. These meetings will also allow group members to adjust priorities before the work is started. The goal will be to complete any outstanding deliverables three days before the actual deadline. The reason behind setting the deadline earlier than the actual due date is so that the team can review the work that has been completed. Task tracking will also be apart of the process work flow and will be done through software such as JIRA, Redmine, or Trello. The task tracking will inform team members on the status of tasks and what each individual member is working on.

Hardware design and construction will follow the rapid application development process similar to Figure 3. The ability to build multiple iterations quickly will help improve the design and allows the software to be tested instantly. Portions of the hardware systems will be built incrementally as well and integrated once they prove to be stable and efficient. The hardware will also be designed using off the shelf components to allow faster iterations. Using agile methodology and rapid application development allows the team to make multiple iterations which is beneficial since the TAs and/or Dr. Alan Wasssyng will be able to give feedback quickly and allow the team to refine the prototype.