Program Family Title [FamNameshould appear in the title —TPLT]

Team #, Team Name
Student 1 name
Student 2 name
Student 3 name
Student 4 name

September 20, 2022

[The CA template is related to the SRS template. Many of the sections are in common. The notes and advice for the SRS template are not reproduced here. Please have a look at the SRS template for advice. —TPLT]

[This CA template is based on Smith (2006). An example for a family of material models is given in Smith et al. (2017). This example is for a physics based family. Often the families will be based on generic numerical techniques, rather than physics. —TPLT]

[A good mindset for thinking about the families is often to think of the family as providing a library of services, as opposed to a single executable. The library of services can be used to build an application that uses a subset of the services, which is like providing the smaller library as a single family member. —TPLT]

1 Revision History

Date	**Version**	**Notes**
Date 1	1.0	Notes
Date 2	1.1	Notes

2 Reference Material

This section records information for easy reference.

2.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	unit	SI
m	length	metre
kg	mass	kilogram
s	time	second
°C	temperature	centigrade
J	energy	Joule
W	power	Watt ($\mathrm{W} = \mathrm{J\,s^{-1}}$)

[Only include the units that your CA actually uses. If there are no units for your problem, like for a general purpose library, you should still include the heading, with the content "not applicable" (or similar). —TPLT]

2.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order.

symbol	unit	description
A_C	$\mathrm{m^2}$	coil surface area
A_{in}	$\mathrm{m^2}$	surface area over which heat is transferred in

[Use your problems actual symbols. The si package is a good idea to use for units. —TPLT] [For the case of a generic numerical library, units will likely not be included. For instance, a linear ODE solver will not know the units of its coefficients. —TPLT]

2.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
FamName	[put your famram name here —TPLT]
T	Theoretical Model

[Add any other abbreviations or acronyms that you add. —TPLT] [Only include abbreviations and acronyms that are actually used. —TPLT]

Contents

3 Introduction

[This CA template is based on Smith (2006). It will get you started, but you will have to make changes. Any changes to section headings should be approved by the instructor, since that implies a deviation from the template. Although the bits shown below do not include type information, you may need to add this information for your problem. —TPLT]

[Feel free to change the appearance of the report by modifying the LaTeX commands. —TPLT]

3.1 Purpose of Document

3.2 Scope of the Family

3.3 Characteristics of Intended Reader

3.4 Organization of Document

4 General System Description

This section identifies the interfaces between the system and its environment, describes the potential user characteristics and lists the potential system constraints.

4.1 Potential System Contexts

[Your system context will likely include an explicit list of user and system responsibilities —TPLT]

- User Responsibilities:
  -
- FamName Responsibilities:
  - Detect data type mismatch, such as a string of characters instead of a floating point number
  -

4.2 Potential User Characteristics

The end user of FamName should have an understanding of undergraduate Level 1 Calculus and Physics.

4.3 Potential System Constraints

[You may not have any system constraints. —TPLT]

[If you need to make design decisions for your family, these decisions will be made here as constraints. For instance, if all inputs will have to use the same file format, this would be a constraint that would be included here. —TPLT]

[You should generally limit the number of constraints, to keep the CA abstract. —TPLT]

5 Commonalities

5.1 Background Overview

5.2 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

-

5.3 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given. [Modify the examples below for your problem, and add additional definitions as appropriate. —TPLT]

Number	DD1
Label	**Heat flux out of coil**
Symbol	q_C
SI Units	$\mathrm{W\,m^{-2}}$
Equation	$q_C(t) = h_C(T_C - T_W(t))$, over area A_C
Description	T_C is the temperature of the coil (°C). T_W is the temperature of the water (°C). The heat flux out of the coil, q_C ($\mathrm{W\,m^{-2}}$), is found by assuming that Newton's Law of Cooling applies (A**??**). This law (GD**??**) is used on the surface of the coil, which has area A_C ($\mathrm{m^2}$) and heat transfer coefficient h_C ($\mathrm{W\,m^{-2}\,°C^{-1}}$). This equation assumes that the temperature of the coil is constant over time (A**??**) and that it does not vary along the length of the coil (A**??**).
Sources	Citation here
Ref. By	IM**??**

5.4 Goal Statements

Given the [inputs —TPLT], the goal statements are:

GS1: [One sentence description of the goal. There may be more than one. Each Goal should have a meaningful label. —TPLT]

5.5 Theoretical Models

This section focuses on the general equations and laws that FamName is based on. [Modify the examples below for your problem, and add additional models as appropriate. —TPLT]

Number	T1
Label	**Conservation of thermal energy**
Equation	$-\nabla \cdot \mathbf{q} + g = \rho C \frac{\partial T}{\partial t}$
Description	The above equation gives the conservation of energy for transient heat transfer in a material of specific heat capacity C ($\mathrm{J\,kg^{-1}\,{}^{\circ}C^{-1}}$) and density ρ ($\mathrm{kg\,m^{-3}}$), where $\mathbf{q}$ is the thermal flux vector ($\mathrm{W\,m^{-2}}$), g is the volumetric heat generation ($\mathrm{W\,m^{-3}}$), T is the temperature (°C), t is time (s), and ∇ is the gradient operator. For this equation to apply, other forms of energy, such as mechanical energy, are assumed to be negligible in the system (A**??**). In general, the material properties (ρ and C) depend on temperature.
Source	http://www.efunda.com/formulae/heat_transfer/conduction/overview_cond.cfm
Ref. By	GD**??**

[In a CA, the TMs often do not need to be refined. However, this is not a rule. In some cases, it may make sense to introduce an IM, or possibly even a GD in between the TM and the IM. —TPLT]

6 Variabilities

[The variabilities are summarized in the following subsections. They may each be summarized separately, like in Smith et al. (2017), or in a table, as in Smith (2006). —TPLT]

[For each variability, a description should be given, along with the parameters of variation and the binding time. The parameters of variation give the type that defines possible values. The binding time is when the variability is set. The possible values are specification time (scope time), build time and run time. —TPLT]

6.1 Assumptions

A1: [Short description of each assumption. Each assumption should have a meaningful label. Use cross-references to identify the appropriate traceability to T, GD, DD etc., using commands like dref, ddref etc. —TPLT]

[Input assumptions will be appropriate for many problems. Some input will have simplifying constraints, and other inputs will not. —TPLT]

6.2 Calculation

[The calculation variabilities should be as abstract as possible. If there are variabilities that are related to imposed design decisions, the system constraints section should be referenced for the relevant constraint. Design constraint related variabilities should be listed separately. —TPLT]

[Variabilities related to data structure choices would go in this section. However, these variabilities are related to design, so they should be separated from the more abstract variabilities. —TPLT]

[Algorithmic variations would go here as well, but as for data structures, they should be separated from the more abstract variabilities. —TPLT]

6.3 Output

7 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

7.1 Family of Functional Requirements

[Since the CA will often be applied to a library, the functionality will not be a single use case. Therefore, this section should summarize the family of potential requirements. A good way to provide an overview of the functional requirements would be to provide multiple use cases on how the library will be employed. —TPLT]

R1: [Requirements for the inputs that are supplied by the user. This information has to be explicit. —TPLT]

R2: [It isn't always required, but often echoing the inputs as part of the output is a good idea. —TPLT]

R3: [Calculation related requirements. —TPLT]

R4: [Verification related requirements. —TPLT]

R5: [Output related requirements. —TPLT]

7.2 Nonfunctional Requirements

[To allow the Non-Functional Requirements (NFRs) to vary between family members, try to parameterize them. The value of the parameter is than a variability. —TPLT]

[An important variability between family members it the relative importance of the NFRs. Smith (2006) shows how pairwise comparisons can be used to rank the importance of NFRs. —TPLT]

[List your nonfunctional requirements. You may consider using a fit criterion to make them verifiable. —TPLT]

8 Likely Changes

LC1: [If there is a ranking of variabilities, or combinations of variabilities, that are more likely, this information can be included here. —TPLT]

9 Traceability Matrices and Graphs

[You will have to add tables. —TPLT]

References

W. Spencer Smith. Systematic development of requirements documentation for general purpose scientific computing software. In *Proceedings of the 14th IEEE International Requirements Engineering Conference, RE 2006*, pages 209–218, Minneapolis / St. Paul, Minnesota, 2006. URL http://www.ifi.unizh.ch/req/events/RE06/.

W. Spencer Smith, John McCutchan, and Jacques Carette. Commonality analysis for a family of material models. Technical Report CAS-17-01-SS, McMaster University, Department of Computing and Software, 2017.

10 Appendix

[Your report may require an appendix. For instance, this is a good point to show the values of the symbolic parameters introduced in the report. —TPLT]

10.1 Symbolic Parameters

[The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance. —TPLT]
[Advice on using the template:

- Assumptions have to be invoked somewhere
- "Referenced by" implies that there is an explicit reference
- Think of traceability matrix, list of assumption invocations and list of reference by fields as automatically generatable
- If you say the format of the output (plot, table etc), then your requirement could be more abstract
- For families the notion of binding time should be introduced
- Think of families as a library, not as a single program

—TPLT]

10.2 First Stage of Implementation

[In this section specify the family member, or sub-family, that you will be implementing. You should specify the value for all of your variabilities, along with the binding time. A tabular representation will probably be the easiest way to convey this information. —TPLT]

10.3 Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

2. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?