Software Requirements Specification for SynthEddy: Simulating Turbulent Flow with Synthetic Eddy

Phil Du (Software)
Nikita Holyev (Theory)

February 6, 2024

Contents

Revision History

Date	Version	Notes
2024-02-01	1.0	Intro, problem, assumptions, goals, I/O and requirements
2024-02-05	1.1	Revisions based on presentation feedback. Finished theoretical models and matrices

1 Reference Material

This section records information for easy reference.

1.1 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
m/s	velocity	metre per second
s	time	second

1.2 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
$\mathbf{x}$	m	3D vector of position in the flow field
t	s	time since the start of the simulation
$\mathbf{u}$	m/s	3D vector of flow velocity at given point and time
$\mathbf{u}_{\text{avg}}$	m/s	3D vector of average flow velocity
$\mathbf{u}'$	m/s	3D vector of flow velocity fluctuation (on top of average velocity)
α	m/s	eddy intensity
σ	m	eddy length-scale
N		total number of eddies
k		index of each eddy, used as a superscript
q_σ		shape function (still need more information)

1.3 Abbreviations and Acronyms

symbol	**description**
A	Assumption
BC	Boundary Condition
CFD	Computational Fluid Dynamics
DD	Data Definition
GD	General Definition
GS	Goal Statement
IC	Initial Condition
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
SynthEddy	Synthetic Eddy Generator
TM	Theoretical Model

Please refer to 4.1.1 for more detailed explanation of fluid related abbreviations and acronyms.

1.4 Mathematical Notation

In this document, the following notation is used:

- Bold font denotes vectors, such as $\mathbf{x}$.
- Superscript k indicates index of eddies, such as $\mathbf{x}^k$ (position of kth eddy).

2 Introduction

When using CFD (Computational Fluid Dynamics) to simulate turbulent flow, it is typical to start with a laminar flow up stream and let it develop into a turbulent flow downstream. However, this is computationally expensive as it need to simulate over a larger region and longer time, before the flow can develop into the desired state.

A potential approach to save computational time is to start with a artificially generated turbulent flow as initial and boundary conditions (IC and BC) for further CFD simulations. Poletto et al. (2013) proposed a method to generate a velocity field of such flow with synthetic eddies, which this software aims to implement.

This project is in collaboration Nikita Holyev, the domain expert in fluid theories. Nikita is a PhD student in Mechanical Engineering at McMaster University, supervised by Dr. Marilyn Lightstone and Dr. Stephen Tullis. He is currently working on the theoretical side of synthetic eddy methods.

If you find yourself with unfamiliar with the these fluid dynamics terminology, please refer to Section 4.1.1 for a brief overview of some background knowledge.

2.1 Purpose of Document

Since the theory behind this software is in a highly specialized field of study, this document helps the software development team to better understand the problem and any related requirements. It also facilitates communication between the software development team and the domain experts to ensure the correctness of implantation.

User of this software can also refer to this document to understand the its capabilities, assumptions and limitations. It also serves as a reference for future maintenance and verification of the software.

2.2 Scope of Requirements

The scope of this software is limited to consider 3D space external flow scenarios only (for now, feasibility of internal flow will be looked into with the domain expert.). It is also limited 1D flow direction with one inlet and one outlet.

As the first step of this project, the software will generate any velocity field based on user provided eddy profiles, regardless of whether they will lead to a realistic turbulent flow. This may change later as theoretical work progresses which allows the software to ensure realistic eddy profiles.

2.3 Characteristics of Intended Reader

The intended readers of this document are expected to have a basic understanding of fluid theories and CFD. If not, a brief overview of relevant information needed to understand this document is provided in 4.1.1.

2.4 Organization of Document

The Introduction section is an intro to this document itself, including reader characteristics and necessary brief background knowledge needed to understand the scope of this project. The General System Description section provides a high-level view of what the user need to know to utilize this software. Detailed problem description and the theoretical behind the software are presented in Specific System Description, followed by Requirements of the software.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

SynthEddy is a software that generates a velocity field for turbulent flow simulations. It can be used standalone or as a external tool for CFD software.

3.1 System Context

The software takes in user provided flow field dimensions, average velocity and eddy profiles and generates a velocity field. The user can then query the software to get the velocity vector at any point of interest in space and time. This query can also be done by CFD software to use the generated velocity field as IC and BC.

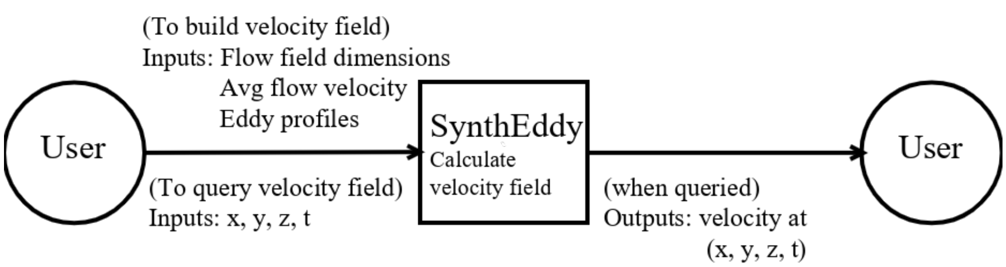


Figure 1: System Context

- User Responsibilities:
  - Provide flow field dimensions and average velocity.
  - Provide eddy profiles. For now, the software will not check if the profiles will lead to a realistic turbulent flow, or suggest any profiles.
- SynthEddy Responsibilities:

- Detect data type mismatch, such as a string of characters instead of a floating point number
- (later) Suggest eddy profiles that will lead to a realistic turbulent flow.
- Generate a divergence-free velocity field based on the eddy profiles.
- Return the velocity vector at any point of interest in space and time.

3.2 User Characteristics

The end user of SynthEddy is expected to have understandings of fluid theories relevant to the use of CFD. They should be able to conduct CFD simulations even without the use of this software.

(for now) They should also have a basic understanding of synthetic eddy theories, detailed later in this document in order to provide suitable eddy profiles.

3.3 System Constraints

SynthEddy should run on the same hardware and operating system that is capable of running the CFD software, which typically is a x86-64 based computer running Windows or Linux. It should have a relatively small CPU and memory footprint, as the CFD software would be computationally expensive and should have priority in system resources.

Certain CFD software may require the use of specific programming languages for any addons/extensions/libraries, but this does not mean the main process of SynthEddy is limited to those languages, as an interface in said language can be created to allow the two to communicate.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models of the synthetic eddy methods used in SynthEddy.

4.1 Problem Description

SynthEddy is intended to alleviate high computational cost/time when simulating turbulent flow in CFD by providing turbulent flow initial conditions and boundary conditions (IC and BC). Since turbulent flow is chaotic and not as easily defined as a laminar flow, in CFD it is typically simulated by starting with a laminar flow and letting it develop over a larger region and longer time before it can develop into the desired state.

4.1.1 Terminology and Definitions

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

- **Fluid Dynamics:** The study of matters that can "flow", including gases (like air) and liquids (like water), and how they interact with other objects (such as an aircraft).
- **CFD:** Computational Fluid Dynamics. The use of computers to simulate the flow of fluids and their interaction with other objects. Commonly used in the both research and design of vehicles, fluid pipelines, etc.
- **Laminar Flow:** Calmly moving flow with no turbulence.
- **Turbulent Flow:** Chaotic flow with many random fluctuations within.
- **Eddy:** A swirling, circular movement of a fluid. A turbulent flow can be seen as consisting of many eddies.
- **Inlet**: Where the fluid enters the simulated flow region.
- **Outlet**: Where the fluid leaves the simulated flow region.
- **IC**: Initial Condition. The state of the entire flow region at the beginning of the simulation.
- **BC**: Boundary Condition. The state of the flow at the boundaries of the simulation region. In CFD, we need to specify the inlet BC.
- **External Flow**: Flow that is not confined by any solid physical boundaries. Think of the flow around an aircraft in open air.
- **Internal Flow**: Flow that is confined by solid physical boundaries. Think of the flow inside a pipe.
- **1D Flow**: Flow that is only in one general direction, which enters the flow region from one end (inlet) and leaves from the other (outlet).

4.1.2 Physical System Description

The physical system of SynthEddy, namely the entire flow field in question, as shown in Figure 2, includes the following elements:

PS1: 3D Flow Field

PS2: Inlet

PS3: Outlet

PS4: Average Flow Velocity

PS5: Individual Eddies

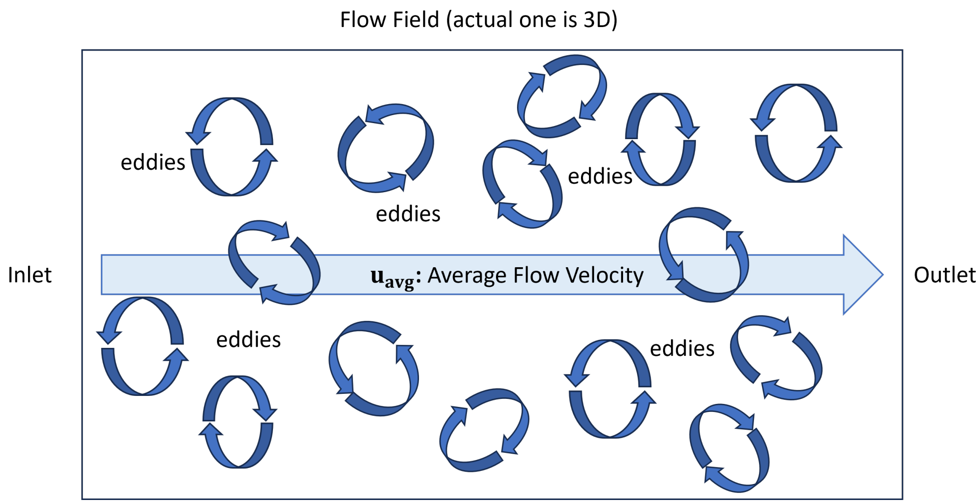


Figure 2: System Context

4.1.3 Goal Statements

Given the flow field dimensions, average velocity and eddie profile(s), the goal (and stretch goal) statements are:

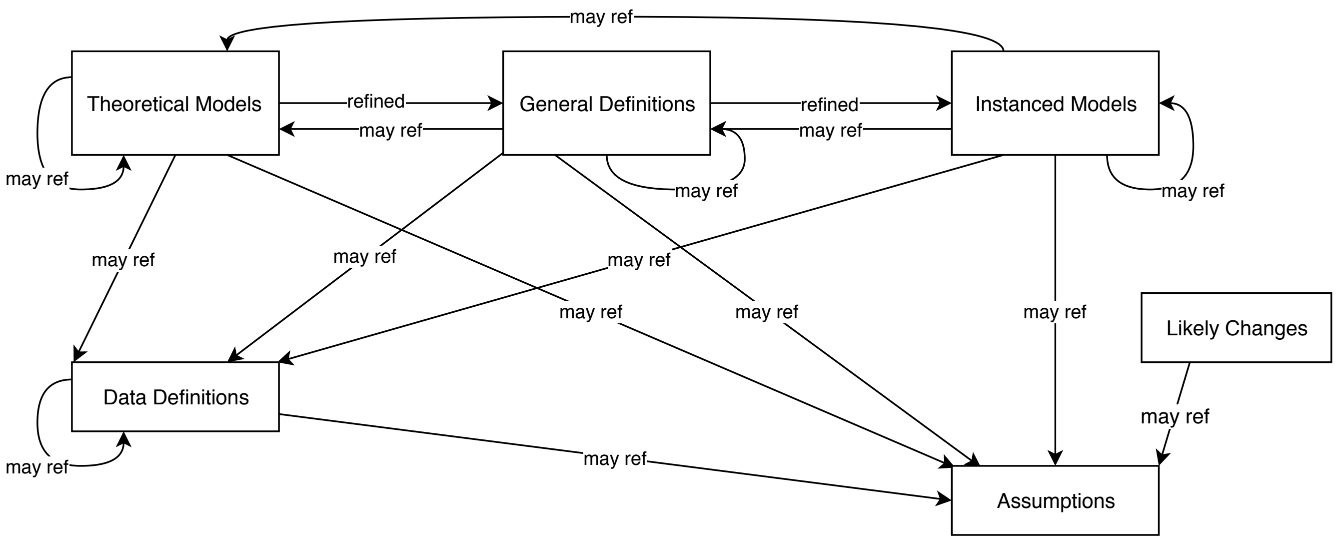
GS1: Generate turbulent flow velocity field with user provided eddy profiles.

GS2: (Stretch) Generate realistic turbulent flow velocity field.

GS3: (Stretch) Allow common CFD software to obtain IC and BC through this program.

4.2 Solution Characteristics Specification

I don't quite understand this section. Are we suppose to develop the solution here or below in Theoretical Models?



The instance models that govern SynthEddy are presented in Subsection 4.4.5. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Types

Inputs:

- Flow field dimensions L_x, L_y, L_z: $\mathbb{R}^3+$
- Average velocity $\mathbf{u}_{\text{avg}}$: $\mathbb{R}^3$
- Number of eddies N: $\mathbb{Z}+$
- Eddy length-scale σ: $\mathbb{R}^+$
- Shape function parameters (still need more info)
- (Query) Position of interest $\mathbf{x}$: $\mathbb{R}^3$
- (Query) Time of interest t: $\mathbb{R}^+$

Outputs:

- Velocity vector at $(\mathbf{x}, t)$ $\mathbf{u}$: $\mathbb{R}^3$

4.3 Scope Decisions

- 3D flow region.
- (Possible removal) External flow only.
- (Currently) No consideration of whether the user provided eddy profiles will lead to a realistic turbulent flow.

4.4 Modelling Decisions

4.4.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the theoretical model [TM], general definition [GD], data definition [DD], instance model [IM], or likely change [LC], in which the respective assumption is used.

A1: The generated flow field is for CFD simulation in 3D space. [DD2]

A2: Flow direction is 1D, with one inlet and one outlet. [DD2]

A3: (Currently) User can provide appropriate eddy profiles. [TM1, TM2]

A4: (Possible removal) The simulation is for external flow. [TM1]

4.4.2 Theoretical Models

This section focuses on the general equations and laws that SynthEddy is based on.

RefName: **TM:SF**

Label: Eddy Shape Function
Number: TM1

Equation: $q_i = \begin{cases} \sigma_i[1-(d^k)^2] & \text{if } d^k < 1 \\ 0 & \text{elsewhere} \end{cases}$
where $d^k = \sqrt{(r_j^k)^2}$

Description: Will have a proper description later.

Notes: This is supposed to be the shape function (q) of a eddy based on location from its center. I think the value means the scale of the flow speed (intensity) at any point of an eddy, and set the speed to be 0 if outside the eddy boundary. There should also be some definition of the spin orientation of each eddy. I am not entirely sure how to understand the shape function equation, or what i and j refer to. I will look into this with the domain expert.

Source: Poletto et al. (2013)

Ref. By: TM2, IM1

Preconditions for TM:SF: A3, A4, DD1

Derivation for TM:SF: Not Applicable

RefName: **TM:VFF**

Label: Velocity Fluctuation Field
Number: TM2

Equation: $\mathbf{u}'(\mathbf{x}) = \sqrt{\frac{1}{N}} \sum_{k=1}^{N} \frac{q_\sigma(|\mathbf{r}^k|)}{|\mathbf{r}^k|^3} \mathbf{r}^k \times \alpha^k$

Description: The equation gives the fluctuation of velocity, the deviation from the average flow velocity at any given point in the flow field. It is the sum of the velocity fluctuation of each eddy at that point, based on the radial distance from the center of each eddy.

Notes: According to TM:SF, at any given position, the fluctuation from most eddies will likely be 0, as any single position is likely outside the boundaries of most eddies in the field

Source: Poletto et al. (2013)

Ref. By: IM1

Preconditions for TM:VFF: A3, TM1, DD1

Derivation for TM:VFF: Not Applicable

4.4.3 General Definitions

There are many forms of refined TM:SF and TM:VFF in the paper. Nikita likely used some form of them his original MATLAB code. I will figure out which one is more suitable to be used in this program and put it here.

4.4.4 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given.

Number	DD1
Label	**Normalized radial vector relative to an eddy**
Symbol	$\mathbf{r}^k$
SI Units	none
Equation	$\mathbf{r}^k = \frac{\mathbf{x}-\mathbf{x}^k}{\sigma^k}$
Description	The positional vector relative to the center of the eddy, normalized by the length-scale (σ^k) of that eddy. $\mathbf{x}$ is the given position (m). $\mathbf{x}^k$ is the center position of the kth eddy (m). σ^k is the length-scale of the kth eddy (m).
Source	Poletto et al. (2013)
Ref. By	TM1, TM2, IM1

Number	DD2
Label	**Center position of an eddy in the flow**
Symbol	$\mathbf{x}^k$
SI Units	m
Equation	$\mathbf{x}^k(t) = \mathbf{x}_0^k + \mathbf{u}_{\text{avg}}t$
Description	As the flow field has a constant average velocity $\mathbf{u}_{\text{avg}}$, the center position of the kth eddy will move with the average velocity over time t. $\mathbf{x}^k(t)$ is the center position of the kth eddy at time t (m). $\mathbf{x}_0^k$ is the initial center position of the kth eddy (m). $\mathbf{u}_{\text{avg}}$ is the average velocity of the flow (m/s). t is time since start (s).
Sources	A1, A2
Ref. By	DD1, IM1

4.4.5 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.4.4 to replace the abstract symbols in the models identified in Sections 4.4.2 and 4.4.3.

Number	IM1
Label	**Velocity field of the synthetic eddy turbulent flow u**
Input	Field construction: L_x, L_y, L_z, $\mathbf{u}_{\text{avg}}$, N, σ, possibly other parameters for the shape function q Query: $\mathbf{x}$, t
Output	$\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_{\text{avg}} + \mathbf{u}'(\mathbf{x}, t)$
Description	L_x, L_y, L_z are dimensions of the flow field (m). $\mathbf{u}_{\text{avg}}$ is the average velocity of the flow (m/s). $\mathbf{u}'(\mathbf{x}, t) = \sqrt{\frac{1}{N}} \sum_{k=1}^{N} \frac{q_\sigma(\lvert\mathbf{r}^k\rvert)}{\lvert\mathbf{r}^k\rvert^3} \mathbf{r}^k \times \alpha^k$ (s) is the velocity fluctuation (m/s). $\mathbf{r}^k(\mathbf{x}, t) = \frac{\mathbf{x} - \mathbf{x}^k}{\sigma^k}$ is the normalize radial vector to each eddy center (m). $\mathbf{x}^k(t) = \mathbf{x}_0^k + \mathbf{u}_{\text{avg}} t$ is the center of each eddy at a given time (m). $q_\sigma(\lvert\mathbf{r}^k\rvert)$ is the shape function (need more info). The above model gives the sum of velocity field from the average velocity and all the eddy fluctuations over time. However, it does not yet have a mechanism to destroy eddies as they move out of the flow field and generate new ones to replaced the destroyed ones. The will be done with programming logics.
Sources	Poletto et al. (2013)
Ref. By	None

4.4.6 Input Data Constraints

Table 1 shows the data constraints on the input output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise.

Table 1: Input Variables

Var	**Physical Constraints**	**Software Constraints**	**Typical Value**	**Uncertainty**
σ	$\sigma > 0$	$\sigma > 0$	0.01 m	0%
$\mathbf{x}$*	$\mathbf{x} \in region(L_x, L_y, L_z)$	$-\frac{L_x}{2} \leq \mathbf{x}.x \leq \frac{L_x}{2}$, same for y, z	1 m	0%
t	$t \geq 0$	$t \geq 0$	5 s	0%

(*) Technically, we can allow queried $\mathbf{x}$ to be outside the flow field and return a velocity of 0 or $\mathbf{u}_{\text{avg}}$, along with a warning.

4.4.7 Properties of a Correct Solution

A correct solution must exhibit a zero-sum velocity fluctuation across the entire field, i.e. velocity field is divergence-free, obeying conservation of mass.

Table 2: Output Variables

Var	**Physical Constraints**
$\mathbf{u}$	$\int_{field} \mathbf{u}(\mathbf{x}) - \mathbf{u}_{\text{avg}} = 0$ (by R4)

5 Requirements

5.1 Functional Requirements

R1: Must verify that the queried point is within the flow region.

R2: Can generate a velocity field given any valid eddy profiles input (IM1).

R3: (Future) Capable of providing a realistic eddy profile before generation.

R4: Generated velocity field must be divergence-free (zero sum of velocity fluctuation, conservation of mass, required by the theory Poletto et al. (2013)) (IM1).

R5: Allow CFD software to interface with this software to obtain IC and BC.

5.2 Nonfunctional Requirements

NFR1: **Accuracy** The accuracy of the computed solutions, namely the how realistically velocity field mimics a turbulent flow, should meet the level needed for being used as IC and BC in CFD simulation. Initially this requirement will not be enforced, until a theoretical to obtain realistic eddy profiles is incorporated.

NFR2: **Usability** The input and output of the software should be easily understandable by those familiar with CFD. Later, it should provide easy setup for integration with CFD software.

NFR3: **Maintainability** The source code should be modular and well-documented. Potential changes to the eddy generation algorithm should be understandable and modifiable by a CFD user if they wish to review the generation method or adapt this software for another theoretical model.

NFR4: **Portability** SynthEddy should be able to run on any x86-64 based computer with Windows or Linux systems.

5.3 Rationale

SynthEddy is based on the proposed method by Poletto et al. (2013) to synthetically generate eddies to form a velocity field that mimics turbulent flow. However, if such method can truly lead to a realistic turbulent flow (given proper inputs) suitable for the purpose of CFD is outside the concern of this software. However, NFR3 makes it possible for a CFD user to review and improve the method or adapt it for another theoretical model if they wish so.

6 Likely Changes

LC1: When theoretical work by Nikita progress, the software will be updated to incorporate the new method to obtain realistic eddy profiles before generating velocity field (A3).

LC2: Possibility to use this software for internal flow will be looked into. This can be valuable as most turbulent flow scenarios are internal (e.g. within a pipe) (A4).

7 Unlikely Changes

LC3: Support for 2D flow field is unlikely to be add, as most computational cost of 2D is significantly less than 3D. Thus, the need for this software is limited (A1).

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 3 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 4 shows the dependencies of instance models, requirements, and data constraints on each other. Table 5 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

	TM1	TM2	DD1	DD2	IM1
TM1			X		
TM2	X		X		
DD1				X	
DD2					
IM1	X	X	X	X	

Table 3: Traceability Matrix Showing the Connections Between Items of Different Sections

	IM1	R1	R2	R3	R4	R5
IM1						
R1						
R2	X					
R3						
R4	X					
R5						

Table 4: Traceability Matrix Showing the Connections Between Requirements and Instance Models

	A1	A2	A3	A4
TM1			X	X
TM2			X	
DD1				
DD2	X	X		
IM1				

Table 5: Traceability Matrix Showing the Connections Between Assumptions and Other Items

9 Development Plan

The first step would be to implement R2 given any user provided eddy profiles, and ensure R4 divergence-free field generation. At this point, the software allows the user to manually query the velocity field at any point within the velocity field (R1).

Later as theoretical work by Nikita progresses, the software will be updated to incorporate the new method to obtain realistic eddy profiles before generating velocity field (R3).

References

Ruggero Poletto, T. Craft, and Alistair Revell. A new divergence free synthetic eddy method for the reproduction of inlet flow conditions for les. *Flow, Turbulence and Combustion*, 91:1–21, 10 2013. doi: 10.1007/s10494-013-9488-2.

Appendix — 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. Which of the courses you have taken, or are currently taking, will help your team to be successful with your capstone project.
2. 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.
3. 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?