

PROJECT PROPOSAL GUIDELINES

Table of Contents

Introduction.....	1
Your Proposal as an Evolving Document.....	2
Document Format & Section Details.....	4
Background and Motivation	4
Project Goal	4
Project Requirements.....	5
Validation and Acceptance Tests.....	7
Technical Design	7
Possible Solutions and Design Alternatives	7
Describing an Initial Technical Design.....	8
System-Level Overview.....	8
Module-Level Descriptions	9
Assessment of Proposed Solution.....	9
Work Plan	10
A Work Breakdown Structure (WBS)	10
Gantt Chart.....	11
Financial Plan.....	12
Feasibility Assessment.....	13
Student-Supervisor Agreement Form	15
Report Attribution Table.....	16
Measurable Objectives.....	17
ECE 496 Draft A Workshop Sign-up Instructions	18
ECE496 Draft A Feedback Form.....	1
ECE496 Draft B Feedback Form.....	1

Introduction

For probably the first time in your undergraduate program, you are required to propose a larger project than likely you've ever done and to carry it out to completion. In teams of two to four students, you will be working on the common project but *individual team members will be required to take on responsibilities for specific work* for which each will be held accountable. Interaction, collaboration and assistance are allowed and expected, but each person will receive an individual mark for his/her work performed in the project.

The first step in this process is the Project Proposal. For an overview, and some hints on common problems at each stage, see [Introduction to ECE496](#).

The Project Proposal is a team document that is much like the PR/PMP (Project Requirement/Project Management Plan), the CDS (Conceptual Design Specification) and some of the FDS (Final Design Specification) you did in first year, APS111 and APS112, all rolled together. It provides the following:

- Background/motivation, goal, project requirements,
- Alternate design solutions,
- Justification for your choice of solution (the “design”),
- A breakdown of the design into parts (modules)
- A set of system tests to prove the final outcome will meet the goal and
- A plan (budget and timeline) to execute the design and tests in time for the Design Fair that is held at the end of the year.

Some of what you were asked for in first year (client analysis for example) are not explicitly asked for here. You are expected, however, to provide enough reasoned information so that both your supervisor and administrator can understand what you are doing, why you are doing it, and the approach you will be taking. So it is up to your team to decide if a client analysis is important and should be included in the proposal information.

Using the information from your proposal, your supervisor and administrator should be able to understand the project well enough that we can help you to detect any serious issues and to address them during or prior to the design review meeting.

A good project proposal takes time to develop and involves the entire team, the supervisor, and the administrator, and you will probably find that it is not easy to generate this document. A popular misconception is that 'all this writing' takes one away from the 'real' design. In truth, producing this document will force you to work on your 'real' design from the top level down, which is often quite unfamiliar for most students and new engineers.

Get as much help as you can from your supervisor. Remember, however, that it is ultimately your project and that you, not your supervisor, are accountable for it. It may be helpful to think of your team as a consulting firm with your supervisor as the 'expert client' who has a good understanding of the problem and the background, and your administrator as your manager who is monitoring your team's progress and performance.

Your Proposal as an Evolving Document

Your project will evolve and change considerably during the writing of the project proposal. To help you develop and think through your work, you will have the opportunity to submit two drafts (A and B) for feedback before the final submission; only the final submission will be marked. The drafts give opportunity for you to submit work, get feedback and improve the final deliverable.

Draft A of the Project Proposal will not be graded, but will be reviewed by and discussed in a meeting between your team and a member of the Engineering Communication Program (ECP) to provide you with feedback for preparing the final version. Attach the [ECE496Y Draft A Feedback Form](#) to the front of your document. When you submit Draft A, you must also sign-up for an appointment at the ECP office. The sign-up instructions are provided in the section, [ECE 496 Draft A Workshop Sign-up Instructions](#). Draft B of the Proposal will not be graded, but will be reviewed by the administrator with your team. The administrator alone will mark the final proposal.

The following table gives the areas to target for Drafts A and B, and what sections are expected for the final proposal. (Every submission should have a cover page. You can, if you wish, add an executive summary, table of contents, conclusion, and references in draft versions before the final submission in order to get feedback.)

TIP: Even if you don't have everything ready for Draft A or B, submit what you have. While there is no "mark" for these, the more you have ready, the more useful the feedback will be, and the more likely you will be to improve your final product.

Section	Draft A	Draft B	Final proposal
Executive Summary			X
Table of Contents			X
Project Description			
Background and Motivation	X	X	X
Project Goal	X	X	X
Project Requirements	X	X	X
Validation and Acceptance Tests		X	X
Technical Design			
Possible Solutions and Design Alternatives	X	X	X
System-level Overview			X
Module-level Descriptions			X
Assessment of Proposed Solution		X	X
Work Plan			
Work breakdown Structure and Gantt Chart		X	X
Financial Plan		X	X
Feasibility Assessment (resources, risks)	X	X	X
Conclusion			X
References	X	X	X
Appendices			
Appendix A: Student-supervisor agreement form			X
Appendix B: Report Attribution Table			X

Document Format & Section Details

The following pages provide specific guidelines for each section of the Project Proposal. In addition, you must follow the [Common Document Guidelines](#) that apply to all the documents you submit for this course.

Background and Motivation

This first section of the Project Proposal is aimed at demonstrating your team's understanding of the technical problem and 'the big picture'. Provide a background, context, and motivation for your project. What makes your project different than what is available? (Note that if the implementation of an existing product is not obvious and is not available, or can be done in an alternate way, then implementing would still make a reasonable project.)

Understanding the problem in the context of the bigger picture requires that you do a literature search, and you should be prepared to put in enough time to build your case. Provide relevant references to original sources of information. References to webpages (like Wikipedia) are generally inadequate, unless they can be justified (e.g. datasheet for components). Wherever possible, reference original sources such as journals, books, and technical standards, and provide complete information in a standard format (See the [Common Document Guidelines](#) for examples.)

Previous Background Work (if applicable)

Many uncertainties about risks are answered in the course of working on a problem. In this respect, teams that have actively worked on their project over the summer have a key advantage, and so should briefly highlight some of the key challenges they have already overcome. Evidence here provides strong support of the feasibility of the remainder of the project. These teams can include some of their previous work as an attachment in the appendix.

Project Goal

The project goal should be summarized in a sentence or two what your project is to achieve. It should be general and non-technical but give direction to the entire project. Two key points are:

- Focus on the desired result, **NOT** the solution or implementation
- Establish some criteria for which the success of the project can be evaluated or verified.

Examples:

GOOD: *The goal of this project is to develop an accurate **circuit model of the cardio-vascular system** (the desired result) that is **compatible with industry-standard circuit simulators** such as PSpice (provides a criteria for success and suggests a possible validation test).*

POOR: *The team will build a vehicle that uses the Arduino board as the controller* (Here the desired result is too vague – this could be a car or plane -- and the implementation is included.)

Projects in industry fall into one of two categories: research or development. Design projects can be similarly categorized:

Research projects are feasibility studies or experiments where the result of the study is not known. For these projects, the **setup** of the study is the functional goal. Such projects may be somewhat harder to define but must meet the same requirements for verifiable project goals. One aid in these cases is to think of what has to be specified to guarantee that another team could exactly duplicate the experiment.

Development projects use established principles to develop an end product.

These projects have hard functional goals but the details of the methodology are left undefined. An example of this type is the building of a microprocessor simulator.

Project Requirements

Provide a list of target project requirements, which will be used to evaluate the success of your project.

Project requirements can be divided into three categories:

- Functional requirements [must-have to fill goal]
- Constraints [legislated or client-enforced]
- Objectives [nice to have. Traded off against cost, time, other objectives]

Functional requirements and **constraints** should be clearly worded in pass/fail terms and in a way that can be **verified**, which implies a corresponding set of verification tests will be needed as discussed in the next section. Project **objectives**, unlike functional requirements and constraints, are not intended to be pass/fail in nature, but are used to indicate the desirable aspects of the final design. The number of requirements depends largely on the project, but at this early stage, the list should not be very long, but enough to capture the essence of your project. The point is to be complete, but not to constrain your design unnecessarily. Here is an example table where the categories of requirements are combined:

Project Requirement	Description
Output: 10.0 +/- .2 V, max. current 10A.	Functional requirement: output specification of power supply
Size: maximum 10 cm x 10 cm x 2 cm	Constraint: This is the size of the available space in the target vehicle
Manufactured cost (in thousands): \$250 or less	Constraint: A higher cost will make the item unprofitable through reduced sales.
Minimize weight	Objective: the lower the better (weight not significant below 100 grams)

Notes on Requirements:

- Requirements should be free of implementation details:
 - **Yes:** ‘The unit shall determine the highest frequency component in the signal’.
 - **No:** ‘The unit shall have an A/D converter and microcontroller, which determines the highest frequency component in the signal.’
- Implementation details *can be included* if they are actually design constraints, imposed either by the client or as a result of limited time or resources.

- Take care to phrase functional requirements and constraints in a way that can be verified (e.g. 'it must be yellow' or 'it must weigh less than 1 kg'). There are some common words/phrases that are often used but which should be avoided. Some examples and ways of circumventing the problems associated with the use of these words/phrases can be found under [Measurable Objectives](#).
- Some parameters can be in multiple categories. For example, for an amplifier, you may have a functional requirement of a minimum gain (e.g. 30dB) and an objective of a gain that is variable (e.g. from 10dB to 30dB).
- Some requirements may not directly relate to the final result but may be necessary for testing or other intermediate goals (e.g. 'create a user interface to monitor and modify the state of the state machine.').
- **For Research Projects:** If you are doing a research project, determining the project requirements may be more difficult. Consider the following questions with the help of your supervisor:
 - What would we need to specify in order to get a research grant?
 - What would we need to specify in order that someone else could repeat this research and get the same results? This is a key requirement for publishing research.

For example, a recent project involved forming optical light channels in plastic using a focused laser to change the refractive index of a path through the plastic. Some of the requirements for this project would concern

- The types of plastic(s) that would be investigated for this use
- The type of laser(s), power setting(s) and focus setting(s) that would be investigated
- The other equipment that would be needed, including the equipment used to test the success of the methods used.
- The types of experiments and test results required to characterize the performance of the fabricated devices.

Clearly, because it is a research project, all of the answers are not known, including how successful the effort will be. However, even negative results are important in research and still indicate a successful project. Also, very important in research will be the risk strategies which you will start to develop in this document. What would you do if the initial plastic selected did not work?

- **For Software Projects:** Evolutionary development methods are acceptable, including agile programming methods, BUT we still expect you to produce a target for your year's work. Your Gantt chart, or equivalent, can show your expected cyclical deliveries of progressive deliveries of features. What is unacceptable is anything that can be interpreted as "we will work in an evolutionary manner, delivering whatever is ready come March."

Setting an Appropriate Scope for your Project

One of the major challenges when setting the project goal and requirements in the early stages of a project is in establishing an appropriate scope for the project. A common mistake is to be too ambitious and to set a scope that is too broad initially, and to be forced to repeatedly redefine the project until the final result has only a small resemblance to the initial proposal. On the other hand, setting a scope that is too limited will affect your evaluation if the project is considered too easy or trivial (e.g. setting up a simple web site). Be realistic: your design project is one of five courses in each semester. As such, each student is expected to spend about a full day a week on it. For team of three, this represents about 500 hours of work over the school year. In industry, this would translate to an effort of about 3 person-months. What can you realistically expect that your team will accomplish in the given time, with the given resources?

Set out a clear scope for your project. Point out the team's key contributions to the final result, and then take advantage of outside resources for the rest. There is no need to implement everything from scratch. Utilize existing resources off the web, commercial software and hardware, resources provided by your supervisor, the Design Centre, etc. to look after aspects of your project that lie beyond the scope of your own work. However, you must make clear **what work is yours (original)** and **what work is borrowed** (and to properly acknowledge or reference that work).

You should also explicitly state what parts of the final system actually lays *outside* your project. For instance, if you are designing the controller for an airplane, you should state that testing your controller in a real airplane is outside the scope of your project and that you plan to use a commercial flight simulator instead for testing and the final demonstration.

Setting the Scope for Research Projects:

Research projects are, by their nature, less likely to follow a specific path. The “project” in the research is the *process*, not the result. For example, consider the team in a previous year that investigated the use of lasers to produce optical pathways in plastic. Their project proposal outlined their initial approach to the problem: The equipment they would use, the constraints they had, the preparation of the samples, the sources of errors and the techniques they would use to discover and to minimize this errors. Over the year the experiments were adapted as they found new information and as they refined and altered their techniques. These changes were expected, but they could be done only because they had a good initial grasp on the problem and potential methods of attacking it. Eventually this team found some very important results, but they were evaluated on the process they used, independent of their success in the experiments.

Research project teams should work closely with their supervisors to first define a clear research question (embedded in the project goal), then draw up a plan on how to carry out the investigation, and provide clear, verifiable requirements for each phase of the investigation that can be used to confirm the team's progress. In this way, it is possible to successfully achieve the project goal in a research project even if no clear answer is found to the original research question.

Validation and Acceptance Tests

In this section, describe how you would validate your final design and prove that it satisfies the project goal and requirements. Consider how you would demonstrate your successful project at the final Design Fair. Alternatively, if you were the paying client, describe the tests you would perform to qualify this product before buying this product. Provide details where possible, including the test equipment, diagnostic software, special arrangements, or test “jigs” that might be required. If you will be doing statistical measures, indicate the number of samples you will test. The point here is to keep your end goal in mind right from the start of the project.

Technical Design

Possible Solutions and Design Alternatives

In this section, you explore and discuss different possible solutions and design alternatives. Exploring possibilities is often neglected by designers eager to start on the first idea that comes to mind. Often, however, the first solution isn’t the best. For instance, you may have in mind an implementation using a keyboard. But when you work back to the requirements, you may realize that the user control can be done as well using a mouse, or through voice commands and hand gestures. The key to designing is coming up

with alternatives, and it is in exploring alternatives that you come to appreciate the inevitable design trade-offs that you will face.

DO NOT ESTABLISH A DESIGN CHOICE, AND THEN THINK ABOUT ALTERNATIVES JUST TO GET THIS DOCUMENT DONE.

Some alternatives may differ only in small variations in implementation while others may be quite different. You should provide enough of an evaluation of each choice to justify your selection of the proposed solution. Provide a preliminary assessment of the different design alternatives in terms of the project goal and requirements you've laid out. Create a comparison table where possible.

You may find that this section and the next naturally collapse into a single section, or that you wish to keep them separate.

Describing an Initial Technical Design

Once you have explored a few design alternatives and their associated trade-offs, you can proceed to propose an initial technical design based on your best design ideas. Do not rush into this stage; even in the final draft of your Project Proposal, you can make the proposed design suitably open to give you the flexibility to define uncertain aspects later. The key is to begin breaking down the overall design into smaller modules and design tasks in order to develop an initial work plan.

System-Level Overview

Begin by summarizing the entire design at the highest level. You may, for instance, explain the principle of operation, the algorithm, or the process flow or stages of an investigation of a more research-oriented project. Make clear if the approach chosen is based on existing, well-established technology or innovative ideas. If your approach is based on existing work, you do not need to provide a detailed explanation; simply point out the approach you wish to use and back it up with a relevant reference.

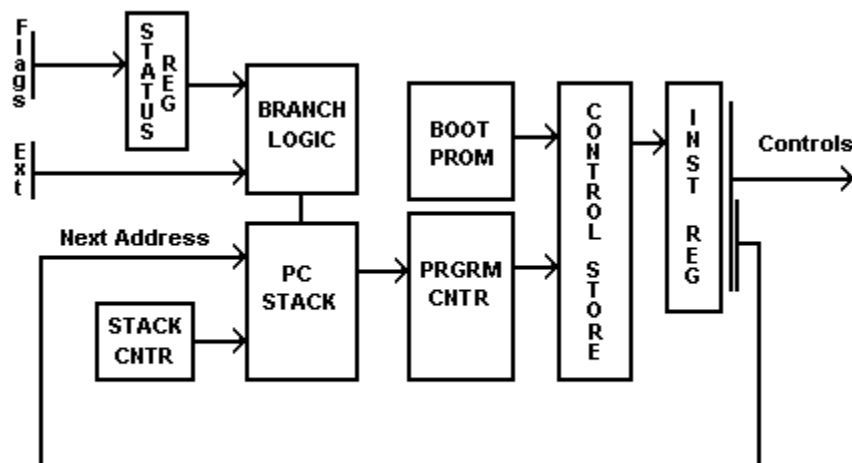


Figure 1 Example of a System Block Diagram

Provide a **System Block Diagram** that breaks down the overall design into its key modules or blocks. Label key inputs and outputs of each module. Modules should have single purpose, be easily described, and should represent a unit that can be designed separately and assigned to an individual team member.

Module-Level Descriptions

Describe or define each module in the system block diagram. Provide enough detail that it is clear how they work with each other in the system. You should also make clear any module requirements where the module affects the adherence of the system to the overall requirements. Also define the inputs and outputs of each module (e.g., RS-232 protocol).

Program Counter (PC) Stack Module Example from Figure 1:

Program Counter (PC) Stack
Inputs: <ul style="list-style-type: none">• Low 16 bits of Instruction Register• 4 bit Stack Pointer• Branch Flag
Output: 16-bit Program Counter Next Address
Function: <p>The Program Counter Stack is configured as a 16-level FILO (first-in last-out) stack. It supports an eight level PC stack where two locations are used for each level. One location is used as the return address for subroutine calls and the second location is used for instruction branching within the current level.</p>

Assessment of Proposed Solution

Comment about the strengths, weaknesses, and trade-offs made in the proposed solution. What reasons led you to choose this solution over some of the others you explored? This section does not need to be long, but ensures that you can provide some justification for your design decisions to date.

Work Plan

This section focuses on managing your project. How do you plan to divide up and schedule the work, and how can you adapt your project and/or work plan if you run into difficulties? Here are the key elements of a work plan:

- A Work Breakdown Structure (WBS)
- A Gantt chart or plan that indicates a schedule of delivery of parts of your project
- A Financial Plan
- A Feasibility Assessment

These elements are described in the sections that follow.

A Work Breakdown Structure (WBS)

A Work Breakdown Structure (WBS) is a hierarchical representation of the major tasks that must be accomplished to complete a design project. It can be presented as a separate chart as shown below or it can be incorporated as the task list in the Gantt chart.

Example: WBS for weekly lawn maintenance

Task #	Task	Jack	Jane	John
	○ Cut Grass			
1	○ Cut front lawn	R		
2	○ Cut back lawn		R	
3	○ Cut sides			R
	○ Trim Edges			
4	○ Trim property boundaries	R		
5	○ Trim around garden		R	
6	○ Trim around trees			R
	○ Trim Bushes			
7	○ Prune side hedges		R	A
8	○ Prune back hedge	R		

R = Responsible, A = Assisting

The above example illustrates the decomposition of work into manageable elements, each of which could be performed by an individual and each of which can be traced to a measurable completion in support of completing the overall job. Note that there is no overlap or duplication of activities.

WBS Checklist:

- ☐ If possible, give some guidelines about how the work is being divided up. For example, you could describe the general role that each team member has on the team (e.g. A is the hardware designer, B the software developer, etc.)
- ☐ Assign the *responsibility* of a task to only ONE person. This does not mean you cannot assist one another. In the end, however, each student is responsible for reporting a clear set of tasks and for showing a clear individual contribution to the project. The person responsible for a task should be the dominant contributor. Break down the task if it is too large for one person to complete, and reassign tasks if necessary. Tasks should take 1-2 weeks of time during the term (assuming that all your courses are equally loading).
- ☐ Each task should have both a number and a description for readability and for easy referencing in the Gantt chart and in future reports.
- ☐ Ensure each task has a clear starting point and end point
- ☐ Every task **MUST** have a verifiable/measurable result. So “Research Microprocessors” is not enough. “Select Microprocessor” would be a correct alternative, since this is closed with the decision “We are using microprocessor xxx”. Creating “testing” as a separate task is NOT acceptable. If “create the communications software” does not include test of the software (perhaps using test data) then it is not a well-formed task.

Gantt Chart

The Gantt chart presents a scheduling of the tasks in a WBS. Continuing with the Lawn maintenance example, below are two sample Gantt charts that illustrate the effect of the number of people on the scheduling of tasks

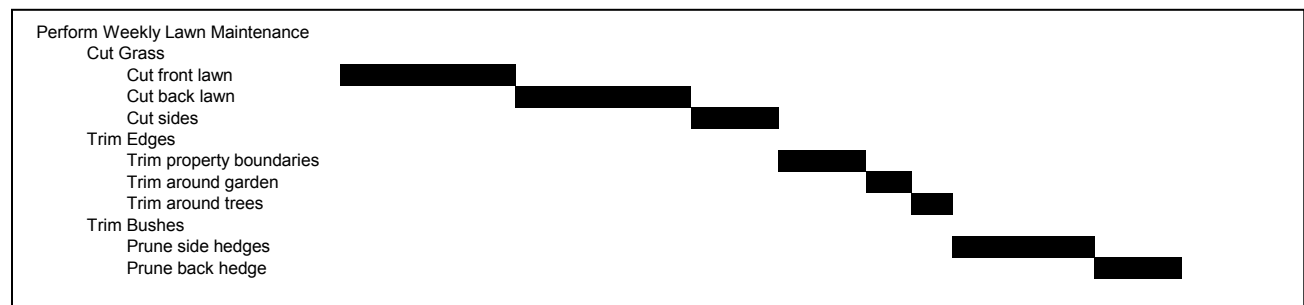


Figure 2: Gantt chart if one person does the Lawn Maintenance.

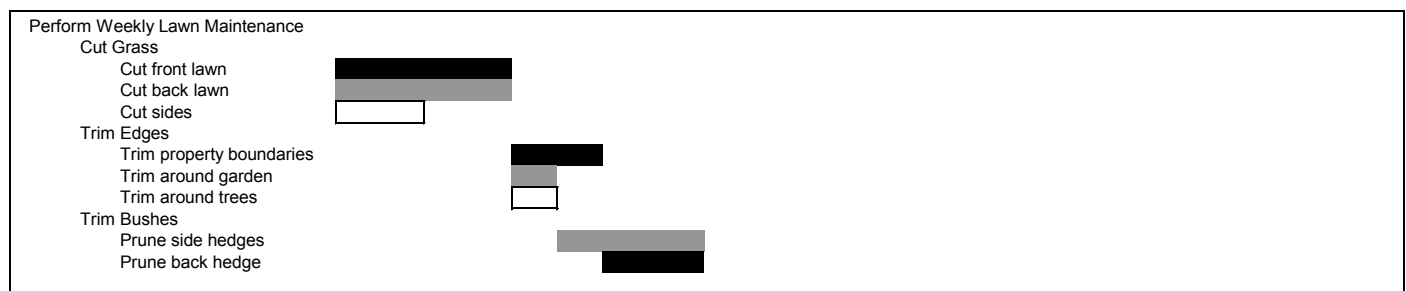


Figure 3: Gantt chart if three people do the same Lawn Maintenance with their own set of equipment. Note that ‘Trim Edges’ cannot be done until the grass has been cut and that having a team of three does not cut the time required by a factor of three in due to the logical dependencies.

The work elements are logically sequenced in time. Some work may be concurrent. Each task generally has predecessor task(s) and successor task(s). To create your Gantt chart, a sample template using Microsoft Excel can be found in the Attachment Library. You can also use a standard scheduling software program such as Microsoft Project Manager.

Gantt Chart Checklist:

- ☐ In order to help you track your progress, your Gantt chart should always show the entire life of the project (i.e., the entire school year).
- ☐ For readability, fit your Gantt chart onto **a single page**. If there are too many tasks on the WBS to fit onto a single sheet, group related tasks into a single overall item or description.
- ☐ Label each task on the Gantt chart with the person that is responsible. Alternatively, you can use a different bar PATTERN for each person. Do NOT use different colours, as they are hard to distinguish in black in white printouts.
- ☐ Apart from holidays, there typically should be no gaps or 'dead time' for any team member where others are working.
- ☐ Every element should have allowance for setup, testing and rework. This is particularly true of the system integration, where modules may need revamping when they don't "fit" together.
- ☐ Avoid cluttering up the Gantt chart with tasks relating to the course deliverables (documents, posters, presentations). Writing reports and giving presentations are understood to be part of your work. If you wish to point these out, you can simply mark the due dates down as milestones.
Is the work duration longer than 3 weeks? If so, the task should probably be broken down into smaller tasks. Generally, smaller duration tasks are more easily measurable and controllable. Long duration tasks tend to "hide" progress or worse "lack of progress".
- ☐ Establish key milestones to serve as checkpoints along the way. A milestone every month or will help you keep track of your progress.

Alternatives to a Gantt chart are acceptable, provided they show a clear path to project completion with measurable, dated module delivery points.

It is expected that things will change. Order of execution may change; tasks may be done faster or slower than first expected; technical problems and their solutions will alter things. However, the Gantt chart should show that you have thought about every step in the proposed process, and have come to realistic (although initial) determinations of time and effort of these steps.

Financial Plan

This section documents the costs of the project, which can include parts, computer hardware, software licenses, rental costs, etc. The Financial Plan consists of a budget table and an explanation of contingency arrangements if the necessary funding for the project is not acquired. You must include your time at a reasonable cost, and should include "free" items that would cost money in industry, budgeted at \$0.

Budget Table

All budgets should be prepared in accordance with the following set of rules:

1. We look to each student to cover at least the first \$100 of expenses from their design project before seeking special funds. Apart from help from their supervisors, the department also has a

special budget to support students with purchases towards their design projects. The budget is limited, but all teams can apply and funding will be awarded based on the strength of your justification for funding. Any rewarded money will not be available until considerably later in the project, so you or your supervisor will have to purchase the items. Make sure you have a contingency plan in place in case you do not receive funding. There are projects available requiring no funding from students.

2. See in example how to differentiate equipment used, used and shared and contractor/service costs.
3. Determine the amount requested from the Design Centre: subtract the net cost from the net contributions.
4. Flag those components/assemblies that you or a team member wishes to keep at the conclusion of the course.
5. Identify the items of higher priority and give a short explanation regarding your contingency arrangements in case you are not successful in receiving the funds or receive only partial support (see below). Example:

"Priority 1) GPS Receiver: we propose to create logic that represents the signal coming from the GPS receiver in case we are not able to obtain funds to purchase the GPS receiver. This will require the following changes to our Technical Solution, Work Plan, and Financial Plan..."

Note:

- A suitable format for the budget request is shown below
- The process for actually requesting funding from the Design Centre is separate from this report. Students should refer to the Budgets and Funding page found in Development Particulars on the course website for detailed instructions.
- For services from a technician or contractor, you should provide a cost breakdown and detailed work plan submitted by the contractor.

See example, Table 1 next page.

Feasibility Assessment

This section is meant to help your team, supervisor, and administrator assess the feasibility of your proposed project. This is not a marketing exercise: try to provide a fair and honest assessment of your proposal, balancing both its strengths and weaknesses. There is nothing wrong with identifying major deficiencies in your project; in some businesses, fewer than one in ten projects results in a commercially viable product. Identifying weaknesses and putting together a plan to address those weaknesses early on is a crucial part of the design process. It also helps your supervisor and administrator in their roles as effective mentors and guides.

Here are some of the key issues you should address in this section. Be brief - a sentence or two is probably enough to cover each issue. Also, these issues do not need to be addressed in any particular order and may be combined or reorganized to flow logically:

Skills and resources:

- What are the key skills, knowledge and resources you need for this project?
- What portions have you already acquired and what portions are currently lacking? How do you plan to obtain what you still need? Examples:
 - From the web: free or open source software, technical standards, expert forums
 - From your supervisor: graduate researchers, lab space and equipment, etc.
 - From the Design Centre (SFB520): facilities for making and soldering printed circuit boards, used hardware from past design projects, computers, test equipment, etc.

Risk Assessment:

In this section, describe the risks that the project could face (risk identification) and how you plan to deal with them (risk mitigation). An example of a risk would be that a particular module might prove impossible to implement, and a corresponding risk mitigation strategy would be attempting to prototype that module first as a feasibility assessment of the whole project. This would be coupled with a ‘fall-back’ plan as to what you would do should this prototype of the module fail. Merely, stating that risks will not occur, or that working harder will mitigate risks does NOT constitute a back-up plan!

This section should not be very long and you should focus on one or two **credible** risks to your project (most likely technical risks but there could be others as well). Minor risks that will cause little disruption in schedule do not need to be addressed. Similarly, you can skip risks that are so general they apply to almost every project e.g. having a team member drop out of school, or losing all their work due to a computer crash. Try to identify risks that are specific to your project. Ask questions:

- What happens to other components and to the project if component X that I’ve designed for my system fails to meet specifications or takes significantly longer to develop?
- Can I change the specifications, demonstrate a lower performance system, or remove some features of my final design and still maintain the essential aspects of my project?
- What if our initial plan to build a real prototype proves unfeasible? Could we demonstrate our design or part of our design using a computer model instead? What would the limitations of this computer model-based prototype be? How would the project goal, requirements, and scope be modified to ensure that the new, redefined, project remains challenging?

Think of ways that you can modify the scope of your project so that you can show some partial success in realizing your project goal by the end of the school year. Note that something that is technically challenging (or difficult to implement) may not necessarily be a significant risk to the project if its failure does not affect the overall project goals or requirements (for example, the task may relate to a project objective for an additional feature, rather than a core requirement).

Risk in Research Projects: Often, a research project will have above-average risk and this section may need to be longer. If the ‘Technical Design’ section describes your intended initial strategy, this section should describe alternate directions that will be taken if experimental results indicate that the initial strategy is no longer what should be done. Here, the ‘risk’ is that the experimental result or some intermediate result not be as expected (and the probability of this could be high) and the mitigation is the determination of the alternate goal or an alternate route to the original goal.

Student-Supervisor Agreement Form

The agreement is available in Word and PDF formats in the Attachment Library. The agreement must be signed by your team and your supervisor, and must be included in the **HARD COPY** of the final Project Proposal. It is NOT required for the electronic copy.

Report Attribution Table

The report attribution table summarizes the contribution of each team member to the project proposal. The table is available in Word and PDF formats in the Attachment Library. Complete the table, showing the initials of each team member in a separate column, and using the abbreviations shown. This sheet must be signed by all team members, and must be included in the **HARD COPY** of the final Project Proposal. It is NOT required for the electronic copy.

Measurable Objectives

The following table illustrates some common problems that students face when they try to come to grips with the meaning of “measurable objectives”. A response to these problems is also given.

Word or phrase that does not constitute a measurable objective	How do you make it measurable?	How do you make yourself credible?
optimal	State project performance goal (target) and how it compares to standard products	Need to make reference to the literature where the numbers or arguments are made. Give a hard number, range of numbers etc.
cost effective	Need to state why it is cost effective. \$ saved by the proposed product etc.	Need to make reference to the publications where the numbers or arguments are made.
improving efficiency	Must be able to qualify and indicate measure of efficiency.	Need to make reference to published material showing current levels.
state-of-the-art	Explicitly state why it is the best and give an example of the current state.	Need to back your claims up with a literature search
larger	Give a number	Back your claim up with evidence quoted in the literature
smaller	Give a number	Back your claim up with evidence quoted in the literature
improved process	How is it improved? What tangible improvements are there over the current process (e.g. power saved xxx, time saved xxx, reliability improved xxx, quality improved xxx; state the performance improvement)?	Back your claim up with evidence quoted in the literature
user friendly	This is a subjective attribute of a user interface like a GUI. It is commonly used to show that some effort will be placed on effective interface design, however it is not.	Do not present this as a measurable objective unless some metric or comparison is included in the work. Polls of peers is OK, but somewhat weak. Better to have hard targets like: No place more than 3 clicks away from the main menu or follows the process recommended for this procedure (and give reference)

ECE 496 Draft A Workshop Sign-up Instructions

Sign up sheets are posted on the bulletin boards of Sandford Fleming B670.

1. Select a time when all your teammates can be at the meeting.
2. Sign up for a time slot: Neatly print the name of your project, the name of a contact person – this should be the person responsible for this milestone – and the e-mail address of the contact person.
3. Write on the front of your proposal: the Session Code, which is two letters and a number (for example, GT4) as well as the time and date of the appointment. Drafts that are missing this information will **NOT** be sorted.
4. Make a record for yourself of the date, time and session code so that you will not forget.

NOTE: Once the appointments are made and the drafts are distributed to tutors, meeting times cannot be changed. Please, arrive at your appointment on time.

ECE496 Draft A Feedback Form

Please fill in your project title, session code, and meeting details, and staple this form to the front of your Draft A Proposal.

Project Title:		
Session Code:		Meeting Date, Time, and Place:

Background and Motivation	Introduces the design problem and its context including state of the art and existing work and technology. Identifies a motivation for the project and the proposed solution.	Comments:
Project Goal	Summarizes what the project aims to achieve. Focuses on the desired result and not the implementation. Identifies main criteria by which the success of the project can be evaluated.	Comments:
Project Requirements	Defines the scope of the project. Functions and constraints are verifiable and objectives are measurable. Requirements are solution-independent and support the project goal.	Comments:
Possible Solutions & Alternatives	Describes possible alternatives and discusses design trade-offs.	Comments:
Feasibility Assessment	Identifies skills, knowledge, and resources required to complete the project. Acknowledges credible risks the project could face and a mitigation strategy for these risks.	Comments:
References	Includes reputable engineering sources. References are complete, cited and support claims made in the document.	Comments:
Writing Quality	Uses clear, concise, and correct language; tone is appropriate for audience and purpose; document is carefully proofread.	Comments:

Specific Areas for Improvement:

ECE496 Draft B Feedback Form

Project No:	Project Title:		
Supervisor:	Section	Administrator	

Required Sections for Draft B	Excellent	Good	Minor changes	Major changes	Poor	Missing Unacceptable
Project Description						
Background and Motivation: A clear introduction and description of the design problem and its context including the state of the art, and existing work and technology. A clear motivation for the work.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Project Goal: Concisely summarizes what the project aims to achieve. Focuses on the desired result and not the implementation. Identifies main criteria for success. Links to the original design problem.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Project Requirements: Can be verified, solution-independent, clearly worded and supports the project goal.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Validation and Acceptance Test: Description of a system-level test that would clearly demonstrate that the project goal has been achieved.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Technical Design						
Possible Solutions and Design Alternatives: Clear description of possible approaches and discussion of design trade-offs.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Assessment of Proposed Design: Clear description of how the solution addresses the technical problem. A clear discussion of the strengths, limitations, and trade-offs of the design.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Work Plan						
Work breakdown structure (circle specific problems) persons not properly assigned to tasks, timeline incomplete	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Gantt chart (circle specific problems) difficult to understand, missing persons assigned to each task	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Financial plan	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Feasibility Assessment: Clear description of required skills and resources; credible risks identified.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Optional Sections (comment if included)						
Executive Summary: A stand-alone, concise, summary of the contents of the document.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
System-level overview: A clear description of the top-level design; system block diagram included.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Module-level descriptions: A clear description of the function, inputs and outputs of the key system modules.	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Overall Quality of Document						
Writing Uses clear, concise, and correct language; tone is appropriate for audience and purpose; carefully proofread	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Content (circle specific problem areas) Organization, logical coherence, substance, references, other (specify):	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Comments:						

The following could be useful to your getting your problem under control, and may be required input for some supervisors:

Problem Statement. Purpose: to guide and aid in the design choices and process.

1. **Problem.** Describe the issue or problem that you are attempting to solve, in layman's terms if possible. No solution should be suggested or implied. On the contrary, it should be possible to think of multiple ways to approach the problem. It is important that the problem is clearly defined and limited in scope.
2. **Design considerations.** These can conveniently be broken down in Functions, Objectives, and Constraints. It is important to think about all the FOCs without any solution in mind.
3. **Client or user characteristics.** Describe all the relevant observations about the intended user(s) of the design.
4. **Additional observations (Optional).** If you can think of additional aspects or ideas that might be considered for the design, but which do not fit into the primary problem as defined above, list them here.
5. **Solution space (Optional).** List, without prejudice, a number of ideas for possible but realistic solutions. A long list is better, even if some solutions may be a bit far-fetched.
6. **Interested parties.** Besides yourselves, your supervisor and administrator, are there other parties with an interest in this project? This may include graduate students, postdoctoral fellows, other professors, other academics or individuals, institutions and industries.
7. **Sources consulted.** Provide a preliminary list of people, institutions and literature that inform on this problem. (A more exhaustive literature review is due later).

Author: Hans Kunov. Draft 1, August 26, 2015

Note: This document is directed primarily to students starting their ECE496 design project at the University of Toronto