



## Product: My Product

### Team: My Group



## Abstract

The abstract should first consist of one sentence describing the intended functionality of your system. It should be followed by a few sentences (100–200 words) summarising the main milestones that will bring your project to a successful completion. This should give the reader a clear expectation of what will be achieved throughout the semester.

## Introduction

This document provides a template for the SDP project plan. This template structures the report into sections, which you are required to use. You can change the subsection headings if you wish. In this template the text in each section will include an outline of what you should include in each section, along with some practical LaTeX examples (for example figures, tables, algorithms). Your document length should be between **six and ten pages**.

You should delete this introduction section (no introduction is required).

## 1. Goal description

This section should start with a paragraph stating, in simple words, the key idea of your system. Describe the problem in one sentence, and use another sentence to describe your solution.

### 1.1. Relevance of the system

If possible you should then provide evidence that the problem is relevant, by referencing published research that demonstrates a need and / or characterizes a potential market for your system. If appropriate, include reference to existing systems that you are taking as inspiration. You should cite the sources (e.g. (Newell & Rosenbloom, 1981)) and add the details to the example-refs.bib file so that the full references appears in the bibliography section.

### 1.2. High-level description

You should then provide the description of your problem in terms of functionalities. A common approach in the software industry is to rely on user stories. Internet is full on resources on the topic and you should investigate it.

## 2. Task planning

In this section you must provide a detailed plan of the tasks to be completed to achieve your goals. The plan must comprise two levels: first a few milestones that correspond to major achievements in the project, then a detailed list of atomic tasks that must be achieved.

### 2.1. Milestones

From the user stories, you should extract the main technical subgoals, i.e., what you need to accomplish to get to the desired final result. For each subgoal you should provide an explicit milestone that states what you should have achieved, by what date, and what evidence you will present to show you have achieved it (e.g. a demonstration of the feature to the experts).

### 2.2. Task decomposition

Each milestone should then be decomposed into a set of “atomic” tasks, taking no more than 20 hours each. Each task should be given a name, a one-sentence-long description, and an estimated time for completion.

You can summarize the tasks in a table, (for instance, table 1, using the `table` environment).

The report must include a Gantt chart, which should clearly identify any dependencies between the tasks. You may find it useful to make a revised version of your plan/gantt chart at key points in the project, in discussion with the experts.

You can include the Gantt chart as a LaTeX figure (such as figure 1), use the `\includegraphics` environment to include an image (pdf, png, or jpg formats), ideally with informative labels added.

To keep your folders clean, it is often a good idea to keep your images in a separate folder. In this example, we’ve put the figures in the `figs/` folder. To include images from different folders, give the relative path from this file. Example: `\includegraphics{figs/image_filename}`.

### 2.3. Resource distribution

The plan should explain how you will deploy your resources - 200 hours per group member over the semester - to achieve your goals. Note you should take into account time required by scheduled sessions (workshops, demo days, final presentations) and time used in planning and presenting (group meetings, report writing etc.).

You should also list the resources you have in terms of skills, equipment, etc. The use of tables is also recommended for

TASK NAME	MILESTONE	ESTIMATED TIME	DEPENDENCY	ROUGH DESCRIPTION
TASK 1	MILESTONE 1	0.5 DAYS	-	DESCRIPTION 1
TASK 2	MILESTONE 1	1 DAY	TASK 1	DESCRIPTION 2
TASK 3	MILESTONE 2	2 DAYS	MILESTONE 1	DESCRIPTION 3

Table 1. Task decomposition for the system

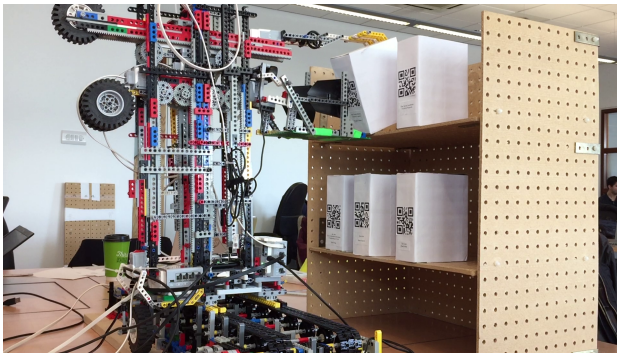


Figure 1. Lego construction: highlight any salient features in the caption

## References

Newell, A. and Rosenbloom, P. S. Mechanisms of skill acquisition and the law of practice. In Anderson, J. R. (ed.), *Cognitive Skills and Their Acquisition*, chapter 1, pp. 1–51. Lawrence Erlbaum Associates, Inc., Hillsdale, NJ, 1981.

this section.

### 2.4. Risk assessment

The report should also contain an assessment of the risks that you anticipate for the project, and contingency planning that you have done to guard against them.

## 3. Group organisation

Finally, how you organise yourselves as a group and plan your work will be key to your success within the System Design project. You should detail the approach that you have taken to group organisation (e.g. specific roles of group members), meetings, communication, code-sharing, task allocation, and progress tracking.

## Submission

This section is to be deleted.

The document should be submitted electronically by one group member using the submit command:

```
submit sdp PP [filename]
```

where [filename] is the name of your project plan file. The filename must be group-[g]-plan.pdf where [g] is the group number. This document should be submitted by a group member nominated for this purpose, and also emailed to the group mentor and to the TA at the time of submission.