START HERE!

Introduction

START HERE!

Application creation can seem overwhelming, especially if you are attempting a project from scratch or for the first time. As you may be familiar with a level of project design and management in your field of work, this guide will focus on the particulars of application development.

We recommend starting from first principles; to begin, try answering the following questions for yourself:

Create a document with your responses to each question - this will be useful for future reference.

Why?

Why do you need this application? What problem(s) does it solve? Why is your application different to anything else on the market?

This may seem like stating the obvious, but answering these questions will allow you to have a clear idea of the purpose of your application. This in turn will help you to avoid feature creep and bloat.

This is the most important question to answer! This is where you define the problem.

Who?

Who is your application designed for? Who will be your ideal user? What level of technological competency do you expect them to have?

Answering these questions will allow you to determine your target audience and give you an idea of how complex/simple to make your finished application.

From this you will be able to describe 'User Personas' - archetypical models of the people who will use your application.

In turn this will enable you to define user roles and their respective access permissions within your application.

What?

What will your application do? What primary function must it serve? What additional features would you like to see? Are these essential or nice-to-have?

You will already have an idea in mind for the purpose of your application, but breaking down the specifics **before** you begin development will save you a great deal of hassle long-term.

From this you will be able to create a feature list - a list of each of the functions of your application. This list should be sorted into priority order before development starts.

How?

How will your application deliver value? How will your application work? How will you manage your data? How much will it cost?

Answer these questions to determine the scope of your project, and anticipate any complexities that might arise during development.

From this you will be able to create a Design Document - a reference guide to the architecture and design of your application. This may become **invaluable** when it comes to future development work on your application, or if a third party wishes to take up development on it!

Where?

Where will your application be used? Tablet, mobile, or laptop?

As Intelastel is a browser-based development platform, your application will mostly be used on any device that supports browsing the web and has an internet connection, that is to say, most devices. Likewise your data sources (spreadsheets and third party integrations) will also need to be web-based.

However, you may wish to consider accessibility and readability as factors when it comes to displaying information to your users.

When?

When do you hope to begin development? When do you you expect to have a working, testable prototype? Will you be able to launch with a minimum viable product (MVP) and then add features and functions? When will you work on updates?

Having an idea of the scope of the project at the start will save you a lot of time, and give your stakeholders an idea of progress and readiness.

From this you can create a Product Roadmap - a document showing the timeline of development, including milestones when key features will (ideally) be released.

Before you start

You will need to prioritise the tasks. This will not necessarily go in the order of most-to-least important, you will need to triage the tasks to work out dependencies and achievability. See the Priorities section below for more information.

Break down each of the features of your application into individual tasks. Set these out in order of importance. You don't need to be specific on each task, a general heading will suffice. If you are familiar with working with a 'kanban' board or project roadmap you could then set out the order of tasks on such a board, thus creating a workflow for yourself.

Any extra features or optional functions should be broken down into tasks as well, but these can then be worked on after you have developed your major feature(s), or included in future development should you find you have the resources.

Considerations

Do you have a group of test users to trial your application once you have achieved MVP? These should be a selection of users from your ideal target audience, with a range of technological competencies.

Alongside budget, marketing and development time, consider the following:

How user-friendly will your application need to be? Will a user guide be necessary to explain navigation and core processes?

Is your data ready to import and use in Intelastel?

Do you have the right permissions to use this data in the Intelastel platform? Consider GDPR requirements.

Do you have an idea of branding? You will need to obtain the colourway and logo from your company's marketing team, or come up with a style yourself.

Priorities

To create a priority order for the list of features and tasks, you will need to *triage* them.

The term 'triage' is defined as "the action of sorting items according to quality".

With regard to designing your application, you will need to determine some key qualities of each feature in order to prioritise their development importance.

This circle looks nice here, doesn't it!

Consider the following qualities for each feature:

Appreciate that the qualities described here are intrinsically linked - the greater the complexity, the more dependencies a feature is likely to have, and the more time/resources it will take to develop.

Complexity

How easy or difficult will it be to implement the feature? Will you need any extra or external resources?

Value

In terms of effort needed to create the feature, versus the gain from the utility of the function. This could be seen as 'time spent on development' against 'time saved using the feature'.

Dependency

What other features need to exist for a given feature to function, and what other features will depend on this one?

These can be considered as key qualities to assess per feature, however there will be other qualities that are necessary to consider, with regards to your business/specialism. It is up to you to determine these and factor them into your planning.

Suggestions

As an approach to visualising your application, you can map out the relationships of the different features of your application. Use a whiteboard, piece of paper, or an online visualisation tool. Include any external or third party components (such as your spreadsheets or databases).

Draw links between each feature to show how they will connect and interact.

From this, you should be able to create a list of dependencies - a list of features and attributes which need to exist for others to function.

This will give you an idea of the order for development of features, as well as enabling you to determine the structure of your application.

Try to create a simple proof-of-concept application as a starting point. This could be a demonstration product that shows the functionality of just one of your intended features.

Watch our tutorial videos and read our how-to guides before starting development. This will allow you to familiarise yourself with the platform.

FINISH!

INTELASTEL

APPLICATION DESIGN - GETTING STARTED GUIDE