Planning a Software Development Project

•••

SQA Professional Development Award in Software Development

Lesson Objectives

Aim: To help in the preparation and planning of a software development project.

- Know a range of techniques that help plan a project.
- Understand the importance of good planning.
- Be able to produce an effective plan to assist in the development of a software development product.

Where To Start: Project Brief

Build a hub for modern interpretations of classics, e.g. noughts and crosses, tic tac toe. Your app will allow users to register an account to track their interactions. Allow them to start new games which they can play against a friend or the computer. They will be able to see a list of the previous games they have played, and a running score of wins/losses. Maybe even a leaderboard of all players across the site, or the facility to start knockout tournaments.

Where To Start: “gather and prioritise”

- Build a hub for modern interpretations of classics, e.g. noughts and crosses, tic tac toe.
- Your app will allow users to register an account to track their interactions.
- Allow them to start new games which they can play against a friend or the computer.
- They will be able to see a list of the previous games they have played, and a running score of wins/losses.
- Maybe even a leaderboard of all players across the site, or the facility to start knockout tournaments.

- **Users**
  - **register account**
  - **track interactions**
- **Game**
  - **start new game**
  - **list of previous games**
  - **running score of losses/wins**
- **Extra functionality**
  - **Leaderboard - all players**
  - **Knockout tournaments**

Where To Start: MVP

What is an **MVP**?

Minimal **V**iable **P**roduct

This is the absolute **minimum functionality** that you must achieve to consider your product viable for production.

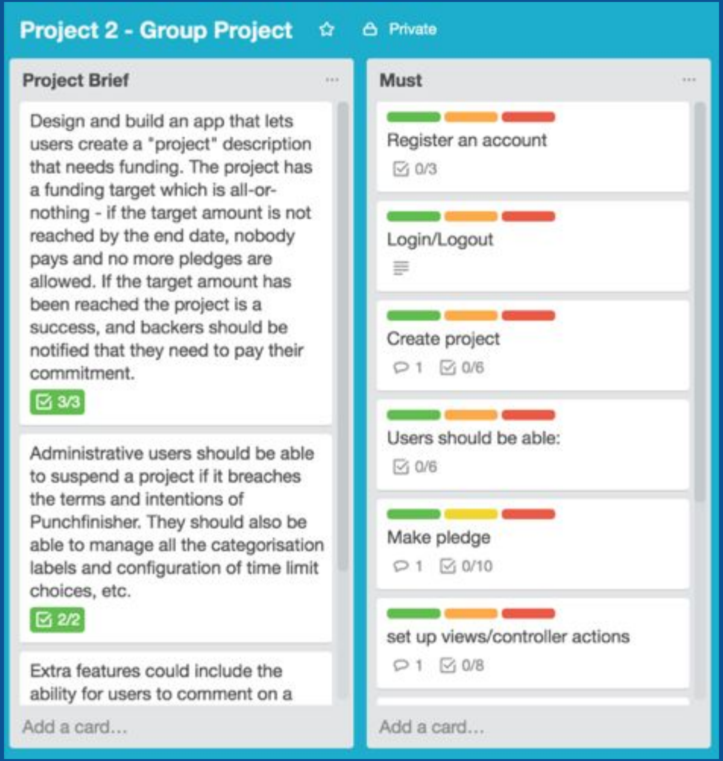
From this point, you can build in extra functionality and make it a little more interesting.

- **Users**
  - **register account**
  - **track interactions**
- **Game**
  - **start new game**
  - **list of previous games**
  - **running score of losses/wins**
- **Extra functionality**
  - **Leaderboard - all players**
  - **Knockout tournaments**

Where To Start: MoSCoW

MoSCoW = **M**ust **S**hould **C**ould **W**ould

- MUST be included
- SHOULD be included
- COULD be included
- WOULD be included



Project 2 - Group Project

Private

Project Brief

Design and build an app that lets users create a "project" description that needs funding. The project has a funding target which is all-or-nothing - if the target amount is not reached by the end date, nobody pays and no more pledges are allowed. If the target amount has been reached the project is a success, and backers should be notified that they need to pay their commitment.

3/3

Administrative users should be able to suspend a project if it breaches the terms and intentions of Punchfinisher. They should also be able to manage all the categorisation labels and configuration of time limit choices, etc.

2/2

Extra features could include the ability for users to comment on a

Add a card...

Must

Register an account

0/3

Login/Logout

Create project

1 0/6

Users should be able:

0/6

Make pledge

1 0/10

set up views/controller actions

1 0/8

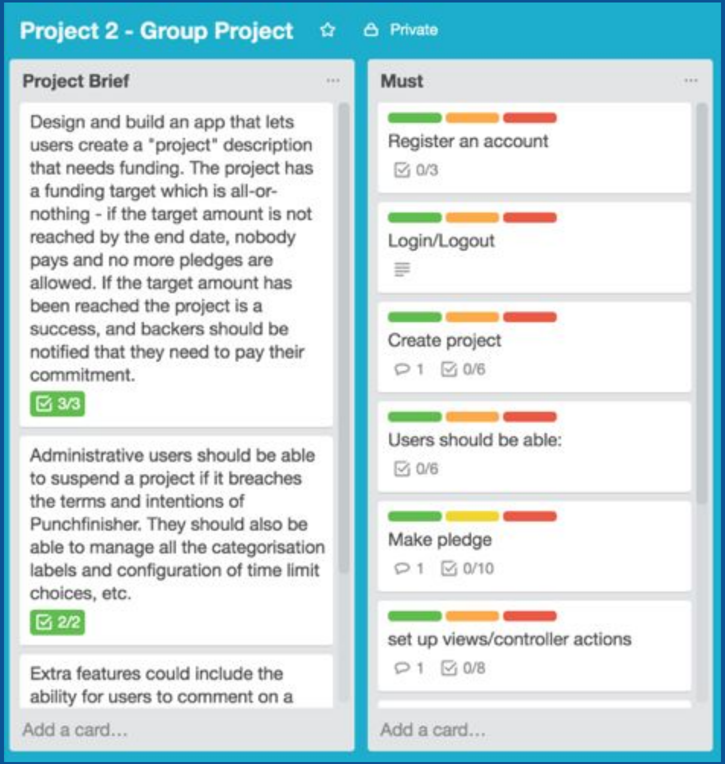
Add a card...

Where To Start: Trello

We can use Trello to record all the information we need for our software development plan.

- Labels, checklists, record diagrams by uploading images, multi-user access, etc.

- https://trello.com/



Design and build an app that lets users create a "project" description that needs funding. The project has a funding target which is all-or-nothing - if the target amount is not reached by the end date, nobody pays and no more pledges are allowed. If the target amount has been reached the project is a success, and backers should be notified that they need to pay their commitment.
3/3

Administrative users should be able to suspend a project if it breaches the terms and intentions of Punchfinisher. They should also be able to manage all the categorisation labels and configuration of time limit choices, etc.
2/2

Extra features could include the ability for users to comment on a

Add a card...

Register an account
0/3

Login/Logout

Create project
1 0/6

Users should be able:
0/6

Make pledge
1 0/10

set up views/controller actions
1 0/8

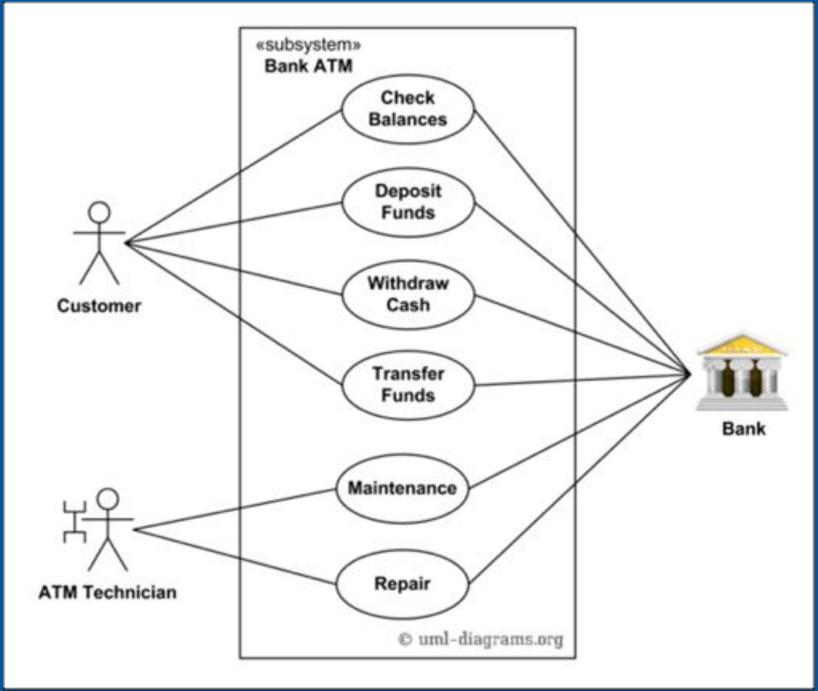
Add a card...

Use Case Diagrams

Use case diagrams give an overview of the usage requirements for a system.

Class diagrams should include:

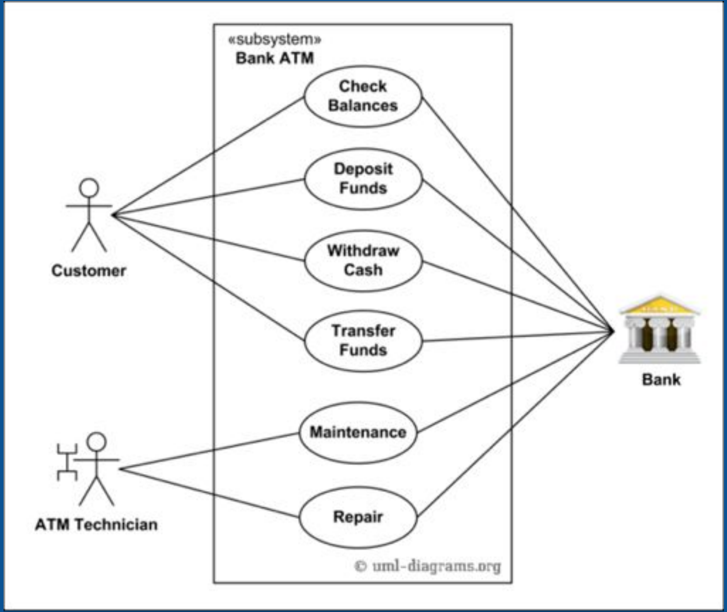
- Use cases - actions that users can perform
- Actors - users of the system
- Associations - connections between users and the actions
- System boundary boxes (optional)



Use Case Diagrams

Create a Use Case Diagram that illustrates a games hub for modern interpretations of classics, e.g. noughts and crosses, tic tac toe.

- Users
  - register account
  - track interactions
- Game
  - start new game
  - list of previous games
  - running score of losses/wins
- Extra functionality
  - Leaderboard - all players
  - Knockout tournaments

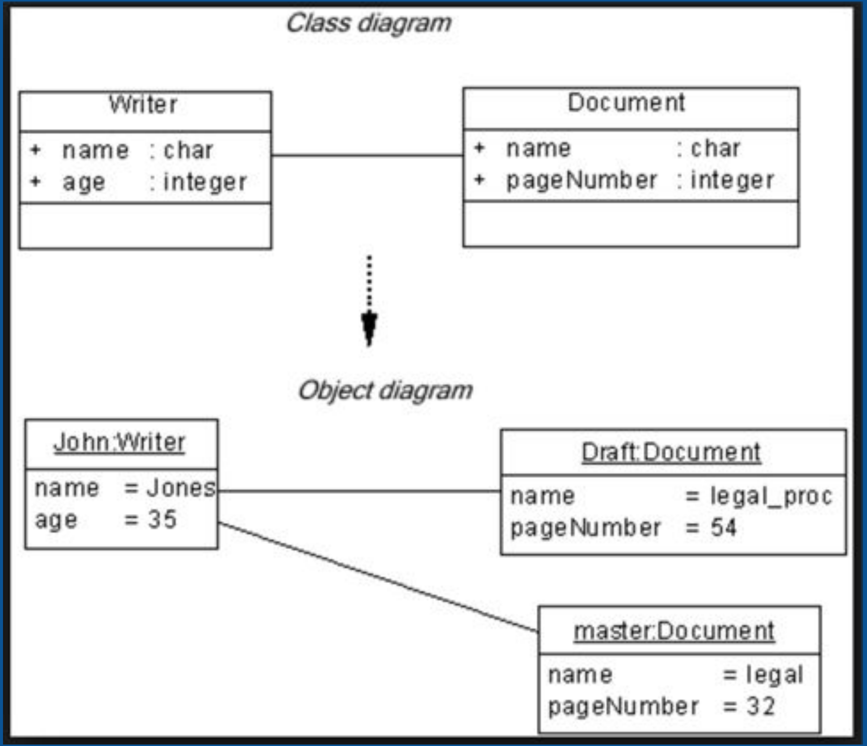


Class Diagrams

Class diagrams describe the structure of a system by showing the system's classes, their attributes, methods and the relationships.

Class diagrams should include:

- Name of each class
- Attributes of each class
- Type of each attribute
- Relationships between the classes
- Any methods that class has

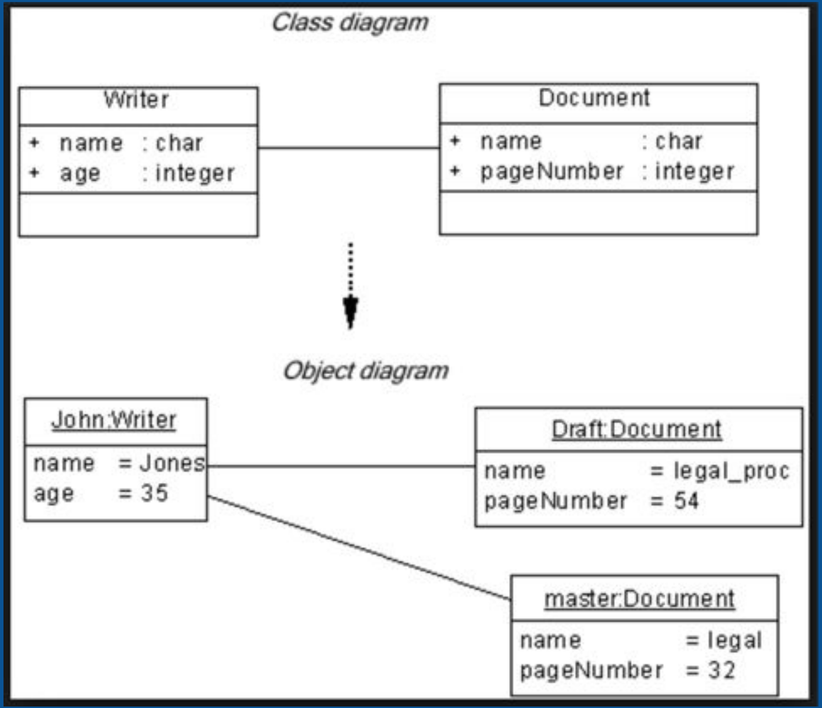


Object Diagrams

Object diagrams provide examples or act as test cases for class diagrams.

Object diagrams should include:

- Name of the class with an instance of that class
- Attributes of each class
- Type of each attribute replaced with an example of that attribute
- Relationships between the classes

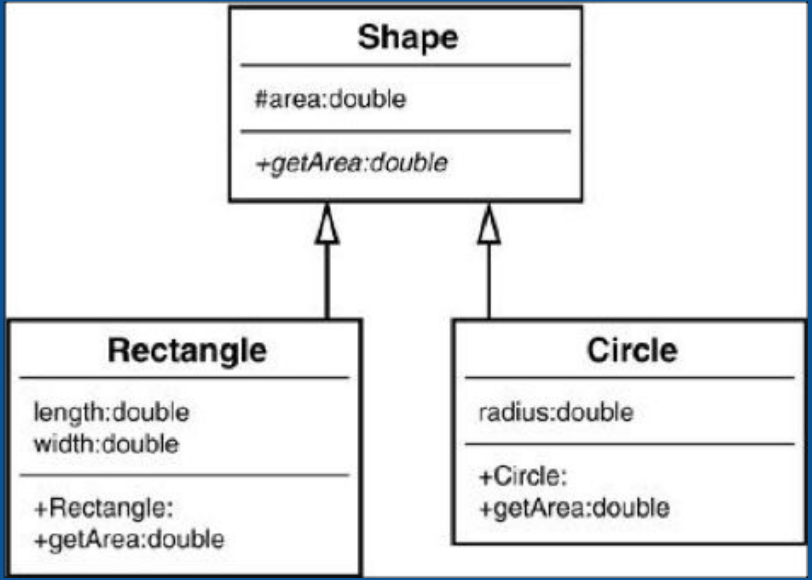


Inheritance Diagrams

An inheritance diagram demonstrates when a child object assumes characteristics of its parent object.

Inheritance diagrams should include:

- Relationships between the classes demonstrated using arrows
- Focus of the relationships is on the flow of information between the classes
- Name of the class
- Attributes of each class
- Type of each attribute replaced with an example of that attribute

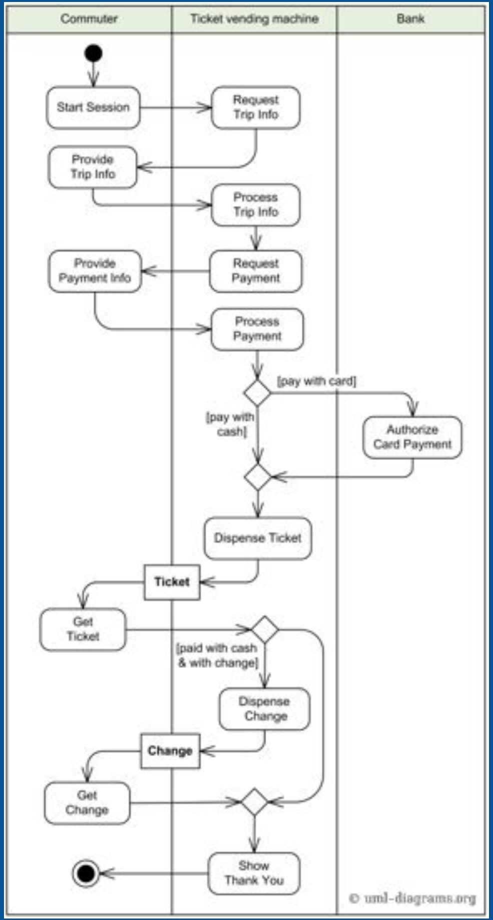


Activity Diagram

An **Activity Diagram** is a flowchart to represent the flow of information from one activity to another activity.

Before drawing an activity diagram, identify the following elements:

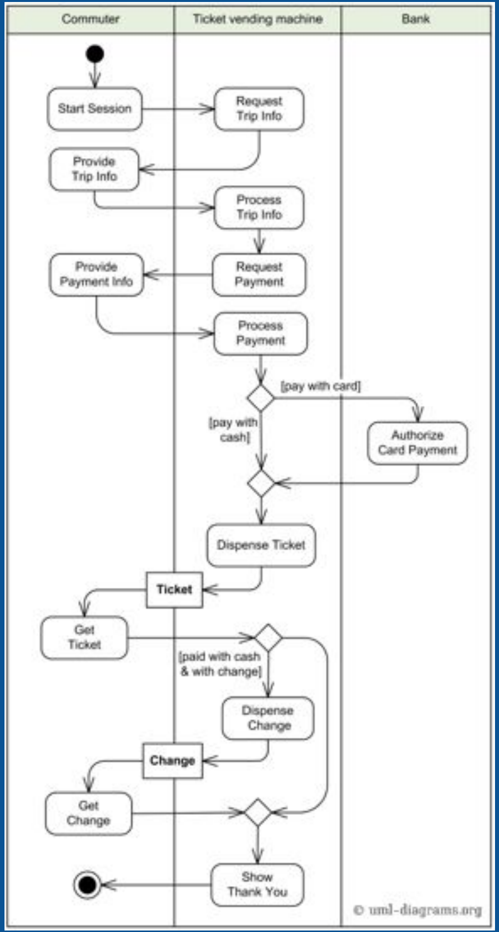
- Activities
- Association
- Conditions
- Constraints



Activity Diagram

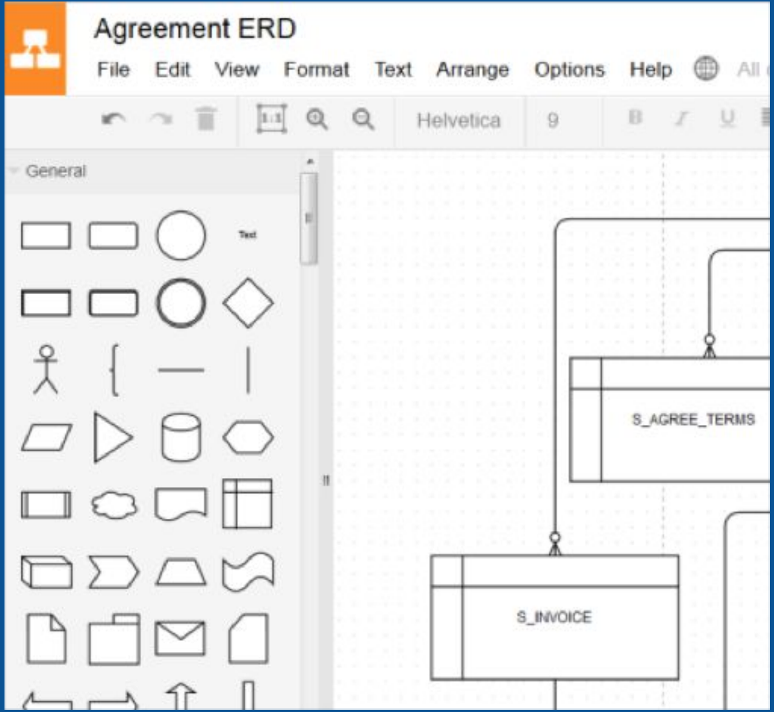
Activity diagrams should include:

- lines with arrows to represent the flow of actions
- rounded rectangles to represent actions
- diamonds to represent decisions
- bars to represent the start (split) or end (join) of concurrent activities
- a black circle represents the start (initial state) of the workflow
- an encircled black circle represents the end (final state).



Diagrams: draw.io

- Online tool that can help you build class or object diagrams.
- Free
- Has a lot of built in functionality.
- https://www.draw.io/



Implementation Constraints Plan

Implementation constraints are things that might affect the project you are working on, in a broader context.

An implementation constraints plan looks at how these constraints may affect the project and consider how they can be dealt with or counteracted.

The plan usually considers the following constraints:

- Hardware and software platforms
- Performance requirements
- Persistent storage and transactions
- Usability
- Budgets
- Time limitations

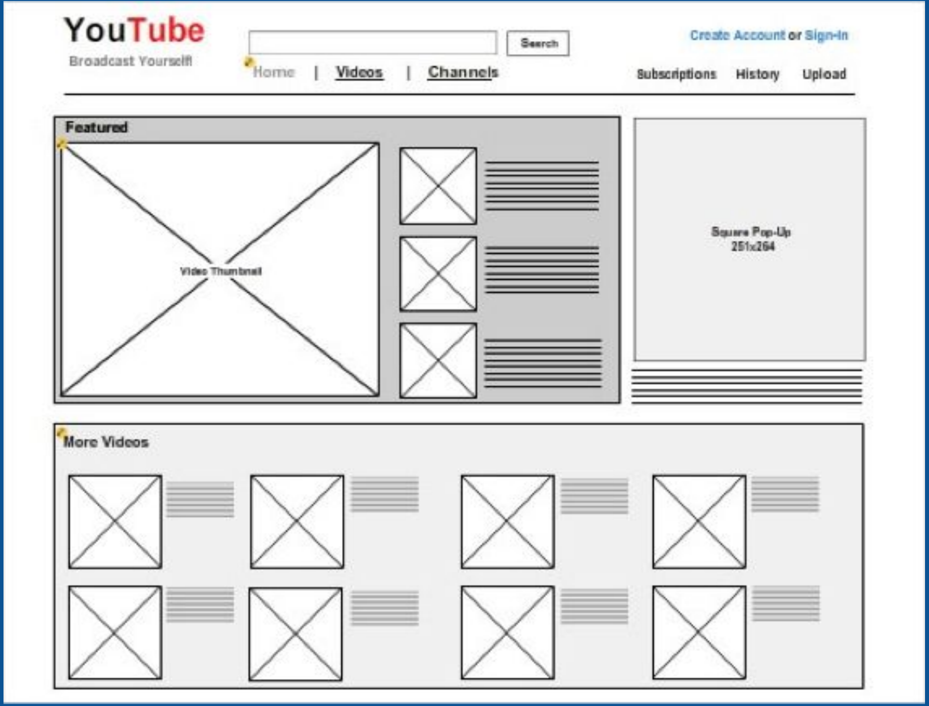
Implementation Constraints Plan

Constraint	Possible Effect of Constraint on Product	Solution
Hardware and software platforms		
Performance requirements		
Persistant storage and transactions		
Usability		
Budgets		
Time limitations		

Wireframes

Balsamiq and NinjaMock are software apps that allow you to build wireframes easily.

- Balsamic - 30 day free trial
- NinjaMock - free



Time Management

- Consider about division of labour and time management

- Once you have made a plan, that is not the end!Good project management means monitoring progress at all stages of the project

- Use AGILE development
  - Morning scrums
  - End of day re-evaluations

Plan a project

For the project brief, complete the following:

- Breakdown the brief and create an MVP.
- Create a Use Case Diagram using your MVP.
- Create a Class Diagram using your Use Case Diagram.

Build a hub for modern interpretations of classics, e.g. noughts and crosses, tic tac toe. Your app will allow users to register an account to track their interactions. Allow them to start new games which they can play against a friend or the computer. They will be able to see a list of the previous games they have played, and a running score of wins/losses. Maybe even a leaderboard of all players across the site, or the facility to start knockout tournaments.

Planning Your Project

Things you should do to help you plan your project:

- Project requirements
- Trello and MoSCoW
- Wireframes
- Time Management

Things you have to do for the PDA:

- Use Case Diagrams
- Object Diagrams
- Class Diagrams
- Inheritance Diagrams
- Implementation Constraints Plan