COMP 2511
Object Oriented Design & Programming

What is User Centred Design ?

Specify the context of use: Identify the people who will use the product (Persona), what they will use it for, and under what conditions they will use it. (Scenario)

Specify requirements : Identify any business requirements or user goals that must be met for the product to be successful. (Use-Cases)

Create design solutions: This part of the process may be done in stages, building from a rough concept to a complete design.

Evaluate designs: Evaluation - ideally through usability testing with actual users - is as integral as quality testing is to good software development.

User Centred Design

- Agile Development philosophy provides value to customers, but in order for our software to be truly successful in the eyes of its biggest critics, we must adopt a more user-centered approach.
- UCD can be applied to the design of anything that has a user—from mobile phones to kitchens.
- The era of **feature-centric development** is coming to an end. Consumers are beginning to realize that more features do not always mean a better product. Quality of experience is far more likely to be a product differentiator than product features
- UCD provides a way to engineer these quality experiences.

Usability Heuristics

1. Visibility of system status
2. Match between system and real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, recover from errors
10. Help and documentation

JavaFX and MVC Architecture

Separation of Concerns Using MVC Architecture

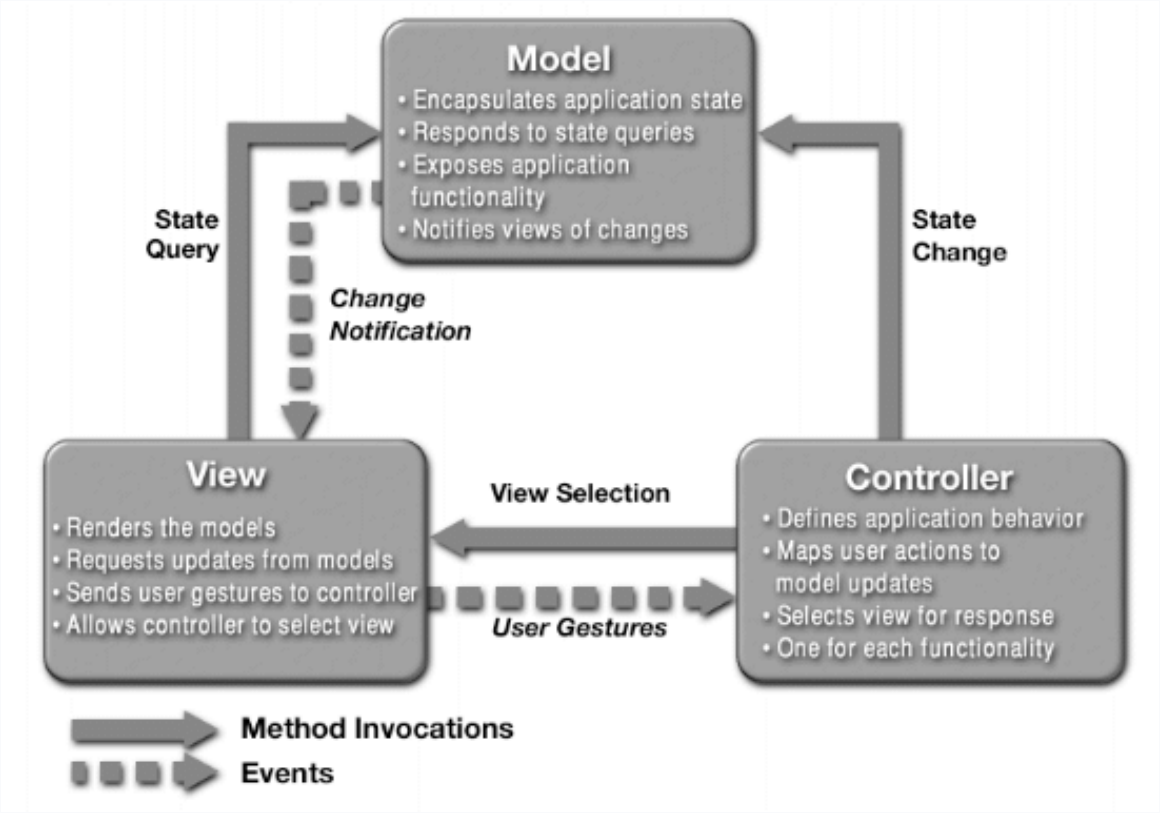


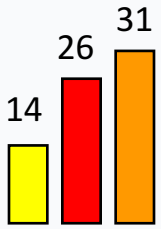
Image http://lkubaski.wordpress.com/2012/12/20/java-se-application-design-with-mvc-with-images/

View

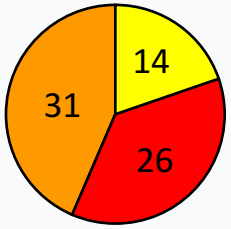
- The presentation layer that provides the interaction that the user sees (e.g. a web page).
- View component takes inputs from the user and sends actions to the controller for manipulating data.
- View is responsible for displaying the results obtained by the controller from the model component in a way that user wants them to see or a pre-determined format (e.g., HTML, XML)
- View can query the model for updates
- It is responsibility of the controller to choose a view to display data to the user.

Model: array of numbers [14, 26, 31]

➔ Different Views for the same Model:



versus



Model (Data)

- Holds all the data, state
- Responds to instructions to change of state (from the controller)
- Responds to requests for information about its state (usually from the view),
- Sends notifications of state changes to “observer” (view) (this “push” behaviour may not always happen)
- The model does NOT depend on the controller or the view

Controller

- Glue between user and processing (Model**)** and formatting (View) logic
- Accepts the user request or inputs to the application, parses them and decides which type of Model or View should be invoked
- Provides model data to the view

Benefits of MVC

- MVC inserts a controller class between the view and model and decouples the two tiers, thereby making the model and view components reusable without modification

- Java FX Lecture demo..