Figure 2.6.: Communication flow of MVC architecture

of each layer. In consequences, the developer can distributed development effort to some extent, so the implementation changes in one part of the system do not require changes to another. For instance, the system may have multiple views sharing the same model, which is easier to maintain , test, and upgrade the multiple system. To add new client, only adding view and controller is necessary. Since the model is completely decoupled from views, it allows lot of flexibilities to design and implement the model considering reusability and modularity. This model also can be extended for further distribution application, which makes the system extensible and scalable.

## 2.2. Framework Popularity and First Frameworks Selection

Nowadays, many presentation development frameworks, such as Java Server Faces (JSF), Wicket, and Tapestry, exist to cover the Presentation Layer. They provide variety of features and architecture enhancements. First step, is to narrow-down the scope of the focused framework based mainly, on framework popularity. The popularity and number of user can determine quality, effectiveness, and assured that the framework has enough qualification to pay attention and considered as one of the research experiment.

Based on the statistic from Google Trends [Rai11b] and Zero Turnaround's Java

EE productivity report 2011 [Tur10], these following seven most widely-used frameworks are selected for further research and analysis:

- Spring MVC (SpringSource.org)[WB05]

- Java Server Faces (J2EE and JSR Standard)[Man05]

- Wicket (Apache Software Foundation)[DH09]

- Seam (JBoss)[All08]

- Struts2 (Apache Software Foundation)[BDS08]

- Tapestry (Apache Software Foundation)[Shi04]

- Stripes (Stripes)

- The existing system, pure JSP and Servlet

## 2.3. General Web Framework Criteria and Frameworks Analysis

Even though, web frameworks give you many benefits, but there are several criteria that should not be neglected. These are the example of those criteria:

### 2.3.1. Testability

Testability is the degree to which an artifact or module support testing in a given test context. Testability cannot be measured directly. In this research paper, the testability of the framework measured roughly by analyzed the architecture and test supported tools provide by the frameworks community. For instance, MVC architecture or other frameworks, which supports clearly separation of each layer, or some frameworks might provide powerful testing API. These two examples provide improvement of testability for the framework.

### 2.3.2. Learning curve

The lower the learning curve is, the better quality the framework is. A good web framework must be not just powerful, productive, flexible, but also need to be easy to understand by the developer, who is new to the framework.