

Model de Casos d'Ús

Temporització: iteració avançada de la fase d'Elaboració (especificacions estables)

Títol del Projecte Plataforma accessible de Gestión Ciudadana con reconocimiento biométrico

1. Diagrama de Casos d'Ús

Construir el diagrama de casos d'ús per als actors identificats

2. Especificació dels Casos d'Ús

Desenvolupar tant l'especificació d'alt nivell com l'especificació expandida dels casos d'ús treballats en aquesta iteració (els indicats a l'activitat corresponent del Campus Virtual)

3. Diagrames de Seqüència del Sistema

Obtenir els DSS associats als casos d'ús que s'estan refinant i/o construint en aquesta iteració

4. Contractes per les operacions

Especificar les operacions identificades en els DSS construïts en aquesta iteració

- **Nombre:** `vote()`
- **Responsabilitats:** El Votante da a conocer que la última opción de voto (partido político) visitada es la escogida para ejercer su derecho a voto. No obstante, el Sistema solicitará confirmación antes de proceder al escrutinio del voto.
- **Tipo:** *VotingKiosk*
- **Referencias cruzadas:** *Funcionalidades: (particulares a cada caso); CU: Emitir voto-e*
- **Excepciones:** ninguna
- **Salida:** pantalla para solicitar confirmación acerca de la opción de voto, antes de proceder al escrutinio del voto.
- **Precondiciones:** Voto electrónico en curso y el votante ha accedido a la información relacionada con una determinada opción de voto. *En caso contrario, **ProceduralException**.*
- **Postcondiciones:**
 - Una instancia *vO* de la clase *VotingOption* ha sido creada.
 - El atributo *party* de *vO* ha sido modificado al valor de la opción de voto seleccionada por el votante actual.
 - La instancia *vO* de *VotingOption* ha sido asociada con *VotingKiosk*.

- **Nombre:** *confirmVotingOption(conf)*
- **Responsabilidades:** Dar por buena la opción de voto electrónico en curso, dando paso a las operaciones de escrutinio del mismo y de inhabilitación del votante para estos comicios (argumento *conf* cierto); o bien no dar ningún paso, permitiendo seguir navegando por el sistema de menús (argumento *conf* falso).
- **Tipo:** *VotingKiosk*
- **Referencias cruzadas:** *Funcionalidades:* (particulares a cada caso); *CU:* Emitir voto-e
- **Excepciones:** Si error de conexión u *Organismo Electoral* caído, indicar el error correspondiente (*ConnectException*)
- **Salida:** 1) Si *conf* es cierto: pantalla de voto emitido; registro del voto emitido en *Scrutiny* para su escrutinio a nivel de colegio electoral; registro en el *e-Census*, vía conexión con el *ElectoralOrganism*, que el votante actual ya ha votado, quedando así inhabilitado (acceso a través del *nif* del votante).
2) Si *conf* es falso: el sistema no avanza, permitiendo seguir navegando por el sistema de menús.
- **Precondiciones:** Voto electrónico en curso; el votante ha seleccionado la última opción de voto visitada, pendiente de confirmación antes de ser emitido. *En caso contrario*, *ProceduralException*.
- **Postcondiciones:**
 - 1) Si *conf* es cierto:
 - La opción de voto actual (*oV*) ha sido transmitida a *Scrutiny* para ser incorporada y registrada persistentemente, quedando así el voto correctamente escrutado.
 - El sistema se ha conectado con el *ElectoralOrganism* para marcar en el *e-Census* que el votante actual ya ha ejercido su derecho al voto para estos comicios (a través del *nif* de la instancia actual de *Voter* se modifica el estado de este votante en el censo electoral, *e-Census*).
 - El atributo *state* de la instancia actual de *Voter* ha sido modificada al valor *disabled*.
 - 2) Si *conf* es falso:
 - La instancia actual *vO* de la clase *VotingOption* ha sido destruida.
 - La asociación entre la instancia *vO* de *VotingOption* y *VotingKiosk* ha sido eliminada.